No. FHWA-RD-77-106

ATIONAL SOFTWARE MODIFICATIONS
HE MOVING MERGE CONTROL SYSTEM
IN TAMPA, FLORIDA

May 1977

Final Report

Document is available to the public through
the National Technical Information Service,
Springfield, Virginia 22161

Prepared for

FEDERAL HIGHWAY ADMINISTRATION

Offices of Research & Development

Washington, D. C. 20590

FOREWORD

This report describes software modifications made to the initial implementation of the computerized green band moving merge freeway ramp control system on I-75 in Tampa, Florida.

The project was carried out jointly by the Florida Department of Transportation and the City of Tampa in cooperation with the Federal Highway Administration. This report was prepared by the Florida Technological University Transportation Systems Institute. The principal investigator was C. S. Bauer. Technical contributions to the project were provided by H. I. Klee, J. G. Bingham, P. R. Boulay, T. A. Risher, A. K. Ehlert, and J. H. Schwarzkopf.

This report is being distributed on a limited basis to selected researchers, a few Washington Headquarters specialists, and NTIS.

Charles F. Scheffey

Charles F. Scheffey
Director, Office of Research
Federal Highway Administration

NOTICE

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA-RD-77-106		

4. Title and Subtitle
Operational Software Modifications to the Moving Merge Control System in Tampa Florida

5. Report Date
May 1977

6. Performing Organization Code

7. Author(s)
Florida DOT and Florida Technological University

8. Performing Organization Report No.

9. Performing Organization Name and Address
Florida Department of Transportation
Traffic Operations Department
605 Suwannee Street
Tallahassee, Florida 32304

10. Work Unit No. (TRAIS)
31C3-553

11. Contract or Grant No.
DOT-FH-11-8285

12. Sponsoring Agency Name and Address
U.S. Federal Highway Administration,
U. S. Department of Transportation
Washington, D. C. 20590

13. Type of Report and Period Covered
Final Report
December 1973 - May 1977

14. Sponsoring Agency Code
T-0231

15. Supplementary Notes
FHWA Contract Manager: S. Tignor (HRS-33)

16. Abstract
In December 1973 a research program on the use of a freeway green band moving-merge ramp control system was initiated between the Florida Department of Transportation, City of Tampa, Florida, and the Federal Highway Administration. Freeway ramp control systems are used to control the flow of vehicles onto the freeway and, thereby, maintain the freeway operations at an acceptable level of service. A green band moving-merge system displays, along the on-ramp, information to the ramp drivers to help them identify gaps and merge easily into the freeway flow. The first test and evaluation of the green band concept was undertaken in Woburn, Massachusetts in 1970. This report describes various traffic operation improvements made to the Woburn green band computer control system. System software changes have been made to minimize the loss of green bands, to stabilize the ends of the moving green bands, and to segment long green bands into shorter, more useable bands. Improvements are also described (1) for preventing the unnecessary surge of ramp vehicles to the merge area when the green band system changes modes, and (2) for making during the stopped mode the system more responsive to ramp and merge area obstructions.

17. Key Words	18. Distribution Statement
Freeway ramps, merging, traffic aids, traffic research, freeway driver, freeway entrances, freeway traffic operations, software	No restriction. This document is available to the public through the National Technical Information Service, Springfield Virginia 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	311	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

PREFACE

This report describes software modifications made to the initial implementation of the computerized moving merge freeway ramp control system on I-75 in Tampa, Florida.

The project was carried out jointly by the Florida Department of Transportation and the city of Tampa in cooperation with the Federal Highway Administration. This report was prepared by the Florida Technological University Transportation Systems Institute. The principal investigator was C. S. Bauer. Technical contributions to the project were provided by H. I. Klee, J. G. Bingham, P. R. Boulay, T. A. Risher, A. K. Ehlert, and J. H. Schwarzkopf.

TABLE OF CONTENTS

METRIC CONVERSION FACTORS

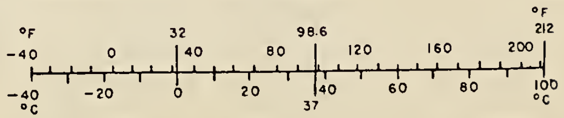
Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



INTRODUCTION AND SUMMARY

The Florida Department of Transportation began a research project with Florida Technological University on March 1, 1976 with the goal of improving the functioning of the control software for the Tampa Green Band Merging Control System. Experience with public operation of the system to that date had identified several areas of operational performance which could possibly be improved or modified to increase the operational benefits of the system to its users.

Meetings of the Technical Advisory Committee (TAC) for the project, consisting of representatives from the participating city, state, and Federal organizations, were held in Tampa on March 11, 1976, August 23, 1976, and January 17, 1977. These meetings provided a forum for identifying desirable software changes for the control system and for subsequently approving the recommendations of the research staff concerning implementation strategies for these changes.

A computer programmer was stationed at the Tampa Control Center during much of the project to code and test the changes comprising the revised system plan. To test the algorithms proposed for smoothing the moving green band displays, simulation models were used to prepare 16 mm films of band activity. These films were used to select and validate the method which was incorporated into the final software revisions.

The original project completion date of December 31, 1976, was eventually extended to April 29, 1977 due to computer equipment problems

at the field site. Formal Software Acceptance Tests were held on April 19, 1977, under the auspices of the TAC and they verified all operational aspects of the control program with special emphasis on the recent modifications. A problem with the system failing to launch green bands from the stop light in the stopped-gap mode of operation during these tests was corrected on April 25, 1977, and the revised system has been in operation since that date. A Technical Evaluation of the new system is to be done by the Florida Department of Transportation with the assistance of the University of Florida Transportation Research Center.

SECTION I: CHANGES NOT RELATED TO BAND PROPAGATION

INITIAL PROJECT SCOPE

The Statement of Work for the project included the following Task Delineation for the control software modifications (an additional Task relating to an extension of the system evaluation activities was undertaken by Florida DOT):

Task G

Modify the Raytheon and Tampa Modified Software pertaining to the propagation of the green band so that (1) the mortality of green bands will be minimized, (2) the ends of moving green bands will be stabilized as the bands move along the ramp, and (3) long bands will be segmented into shorter bands to provide the sensation of moving bands.

Task H

Investigate the suitability of the new, proposed display algorithms, prior to selection for field implementation, by using the simulation developed by Bauer. This investigation will consider the impact of detection reliability and sensor configuration on the performance of the new display algorithm.

Task I

Modify the Raytheon and Tampa Modified Software to prevent the release of a surge of vehicles onto the ramp when the system goes from a stopped mode to the moving mode.

Task J

Modify the stopped mode logic used in the Raytheon and Tampa Modified Software so as to be more responsive to ramp and merge area obstructions.

Section II will cover the preparation of movies used to evaluate band propagation algorithms developed to meet the requirements of Tasks G and H. Section III will discuss the structure of the final band propagation algorithm selected for field implementation. This section will review 1.) changes made to the control software to meet the requirements of Tasks I and J, and 2.) other changes which were made at the request of the TAC to provide additional operational improvements to the green band system in Tampa.

Modifications Implemented for Task I: Improvements to Mode Change Logic

To prevent the release of a surge of vehicles into the merge area of the ramp when the system transitions from stopped gap (SG) mode to moving mode (MM), the following procedure was inplemented in the Green Band Status Subprogram:

Allow a transition from SG to MM only if all of the following conditions are met:

1. Normal velocity and volume criteria for this transition are satisfied.
2. The system is not "masked" at the candidate transition time.
3. Ramp sensors R6B and R7 have been clear of traffic for a fixed period of time (in the range of 0-60 seconds, operator-changeable from the system teletype keyboard.)

Modifications Implemented for Task J: System Response to Ramp and Merge Area Obstructions

To reduce vehicle queueing in the ramp and merge areas downstream of the traffic lights on the ramp, a procedure was designed for implementation in Green Band Status to meter vehicles every 20 seconds from the lights in the SM and SG modes if either or both of the following conditions are true:

1) Any of the detectors R1 through R5 have been occupied for more than some fixed time period (operator controllable from the console).
2) Either of the merge area detectors M1 or M2 have been occupied for more than some fixed time period (also operator controllable from the console.)

Modifications Implemented for "No Vehicles On Ramp" Condition

The Tampa green band system uses an "open-loop" control scheme in the sense that vehicle movements on the ramp are not used in the display algorithms to determine band placements. In the earlier Tampa system, as a consequence of this design, bands were displayed in the moving mode at all times when adequate freeway gaps were available, even when vehicles were not actually on the ramp. To eliminate this condition, band displays are now normally kept off, and a software timing loop started by an R-11 activation is now used to turn on band displays for a period of time adequate to allow the triggering vehicle to clear the ramp. (This time is operator controllable from the console and is nominally 20-30 seconds.)

This feature should help contribute to driver acceptance of the system, as the displayed bands will now appear to be responding to individual vehicles entering the ramp. In addition, a slight reduction in the power consumption used by the green band should be expected, as the bulbs will not be in the full on condition as much as they were in the past. (The "quick-start" ballasts of the green band display have roughly the same power consumption when the bulbs are off as when they are on to keep the bulb elements ready for operation.)

Modifications Implemented for Improved Capabilities In Sensor Performance Monitoring And System Response To Sensor Failures

To provide a capability for identifying field sensors which are malfunctioning (e.g., "stuck-on", stuck-off", or "chattering"), and to provide appropriate safety measures in normal unattended system operations, the following features are now implemented in the Operator Monitor and Fault Monitor Subprograms:

1) The system will now automatically print a numerical count of all sensor activations since the last initialization and since the last print interval (operator changeable time, normally one hour.) These counts may be used to identify sensors which are not in acceptable agreement with other sensors in their area. This feature can also be requested manually from the teletype keyboard at any time by pressing the "S" key. The "I" and "U" keys can be used to inhibit and uninhibit (e.g., allow) the automatic print feature as desired.

2) A "critical sensor" monitor is now implemented for the following sensor groups and their associated system functions. If detector state changes are not uniform for a group within a fixed time period, i.e., all detectors "on" at least once or, no detectors "on" for a period of time (operator changeable, nominally 3-5 minutes), a fault message will be printed on the teletype, and the system will be automatically brought to a safe full-off condition and the computer halted.

Function	Critical Sensors Monitored
yield sign control	R3, R4
freeway traffic prediction	F1A through F4B, inclusive
traffic signal control	R6B, R6A
merge area monitoring	M1 through M5, inclusive
power off masking	R11
volume counting	R1, R2, R3 (2 out of 3 must be active)

Modifications Implemented For Redundant Sensor Activation of the Yield Sign

To provide an additional safety margin for the control of the YIELD sign on the ramp, a new control procedure using both R3 and R4 has been implemented. A vehicle now has to have a green band at both R3 and R4 to keep the yield sign from being activated.

Modifications Implemented To Provide More Reliable Data For Modal Switching Algorithm

For the first generation Green Band System in Tampa, system determination of freeway speed and combined freeway/merging volume was based on sensors F1A/F1B and FOA, respectively. These parameters are used in the automatic mode switching logic in the Green Band Status Subprogram.

To provide more reliability in the measurement of velocity and volume, the system now obtains these values from the following multiple-sensor operations:

$$\mathrm{VBAR} = \frac{\left[\mathrm{V(F1)},\ \mathrm{V(F2)},\ \mathrm{V(F3)}\right]}{2}$$

$$\mathrm{VOLUME} = \frac{\left[\mathrm{VOL(F1A)},\ \mathrm{VOL(F2A)},\ \mathrm{VOL(F3A)}\right]}{2} + \frac{\left[\mathrm{VOL(R1)},\ \mathrm{VOL(R2)},\ \mathrm{VOL(R3)}\right]}{2}$$

where $[\ ]$ denotes the sum of the closet two measurements.

Modifications Implemented To Provide An Alternate "CHECK-IN" Logic at the Stop Bar

Operational experience with the earlier Tampa system in the stopped gap (SG) and stopped metering (SM) modes has shown that many vehicles have stopped upstream of a point on the ramp close enough to trigger the activation of R6B, the nominal "check-in" detector for service demand at the traffic signal. In such instances, the light remains red and queueing

can develop until the lead driver moves from a stopped position.

To provide an alternative system response for this condition, a new algorithm has been added to the Green Band Status Subprogram to initiate the light cycle logic as follows: If the traffic light has been indicating red for a period of 15 seconds or more, and R7 has been activated within the last 15 seconds and R6B has not, a new cycle will be initiated as if R6B had been triggered.

This modification should enhance driver confidence in the system by having it respond to the special needs of drivers not familiar with the system and thus who are possibly expecting a fixed time cycle at the light.

SECTION II: PRODUCTION OF THE COMPUTER GENERATED GREEN BAND MOVIES

The work described in this section can be considered to be three relatively independent tasks. The first task involved changing the simulation model (ref: Bauer, 1975) to obtain the data required to draw a movie frame. The work involved making changes in the existing simulation code and the addition of new subroutines.

The second task was to develop a program to process the frame data and draw the pictures on the graphics terminal. Since a time-sharing graphics terminal was to be used, the program execution had to be controlled by the operator to allow frame selection and photographing.

The third task was the execution of the drawing program and the photographing of the movie frames. Since this phase of the project was to allow the comparison of green band control techniques, it was necessary to make several movies using the various band control techniques.

Figure 1 is a diagram of the process involved in the making of a computer generated Green Band Control Program Simulation movie. The processed sensor data serves as input to the green band simulation model which creates the frame data. The drawing program then uses the frame data as input to draw each frame of the movie on the graphics terminal, where it is photographed on a single frame basis using a 16 millimeter movie camera. The film is developed and the movie process is complete. The movies are

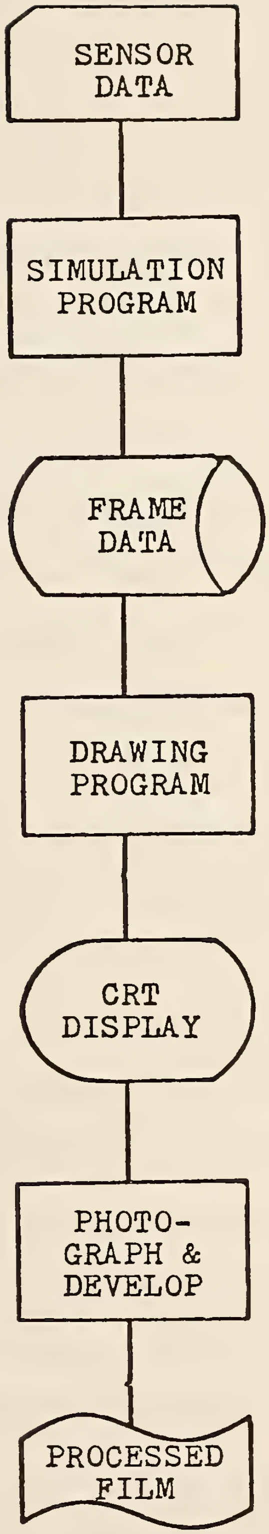


Figure 1. The computer generated green band movie process.

are then available for analyzing the effects of changes in the Green Band Simulation model.

The remainder of this section provides a detailed description of the work accomplished in the completion of the tasks. Listings of the Computer Programs discussed herewith may be found in Appendix A.

Generation of the Frame Data

The data required to draw a picture of the state of the freeway and the band display include the positions and types of the vehicles along the freeway, the status of the green band display, and the current simulation time. To acquire the data at a time interval corresponding to the frame exposure rate of the film required modification of the existing Green Band Simulation program.

The modification involved the addition of an interrupt to the simulation main program, changes in the GBUD subroutine, and the addition of two subroutines.

The interruption of the simulation to allow the output of the current state was provided by a cycle counter and the use of IF statements to test the number of cycles since the last frame data output. Each operating cycle in the simulation represents 2 milliseconds of the green band system time history. The number of cycles between interrupts for a frame output is dependent on the film speed to be used. The following equation provides the value for the number of cycles between interrupts for any film speed:

$$\mathrm{CBI} = \left(\frac{1}{\mathrm{FS}}\right)/0.002,$$

where CBI is the number of cycles between interrupts, rounded to the nearest

integer, and FS is the film speed in frames per second. For rates of 18 and 24 frames per second, the number of cycles are 28 and 21 respectively.

The changes in the GBUD subroutine involve the assignment of numerical values to represent the status of the green band and the division of the green band into 4 foot segments to represent the 4 foot fluorescent lights used in the actual system. Each 4 foot segment was assigned either a value of one or two corresponding to an off or on status respectively. The division of the band into 4 foot lengths was accomplished by changing appropriate constants involved in the GBUD subroutine statements.

The first of the two subroutines added to the simulation, XVEH, uses two methods to provide the current locations of the freeway vehicles. Figure 2 is a simplified flow chart of the XVEH subroutine. The RAWDAT lists created in the ISRINI subroutine are examined on a sensor by sensor basis starting from the one closest to the ramp, F1, to the fourth sensor, F4. Lists for sensors F5 through F7 are not examined since they are physically located before the region of the freeway displayed on the graphics terminal. The display begins 57 feet downstream of sensor F4 and extends 1,023 feet along the freeway.

The XVEH subroutine begins by initializing vehicle index counters and setting all vehicle flags to zero. The vehicle flags are used to indicate whether a vehicle of a particular number has been processed during each call of XVEH. Since the sensors are examined from the closest location to the merge point to the farthest, the flags prevent a vehicle location from being recorded from more than the most recent sensor entry.

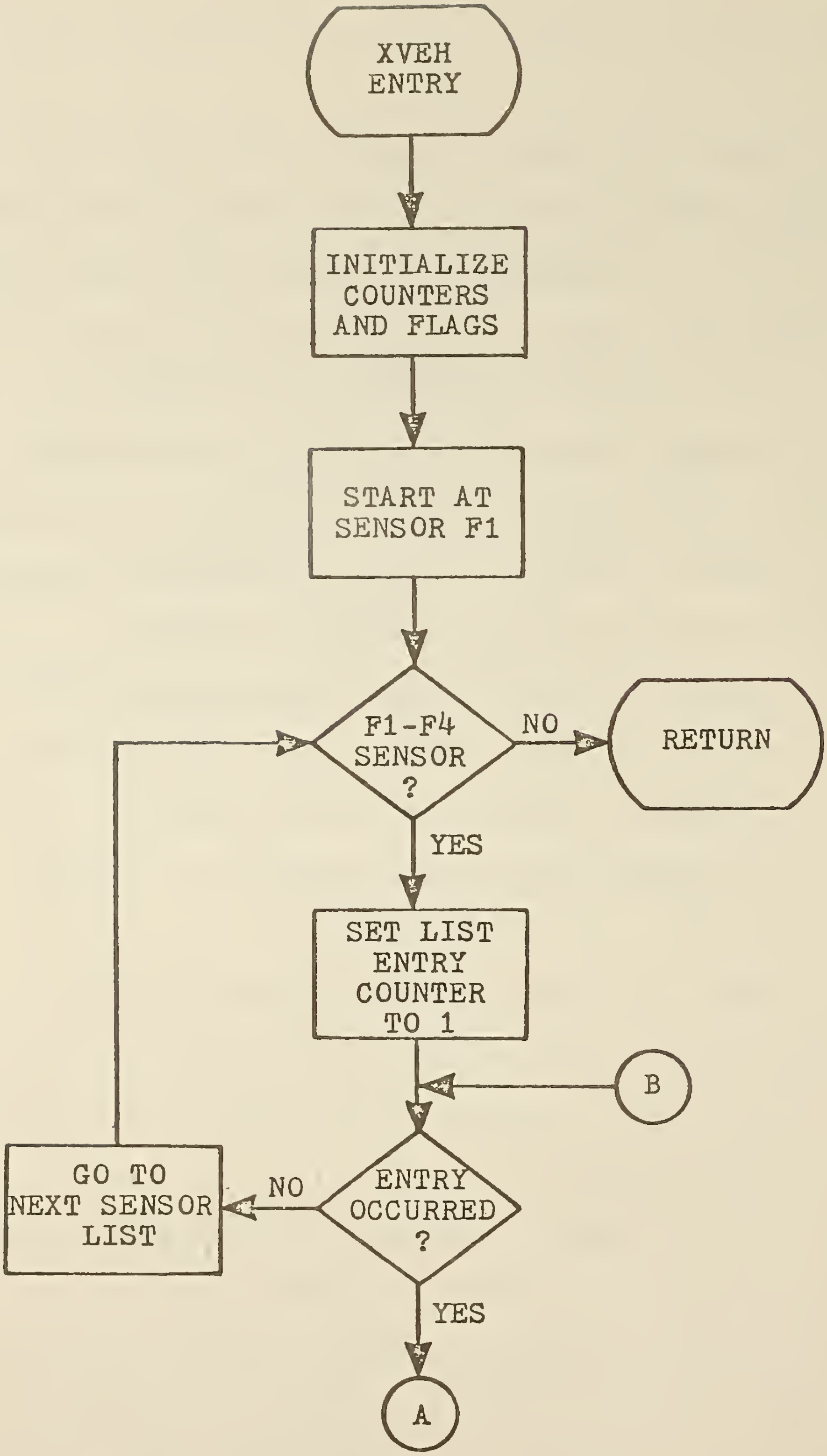


Figure 2. Flow chart of the XVEH subroutine.

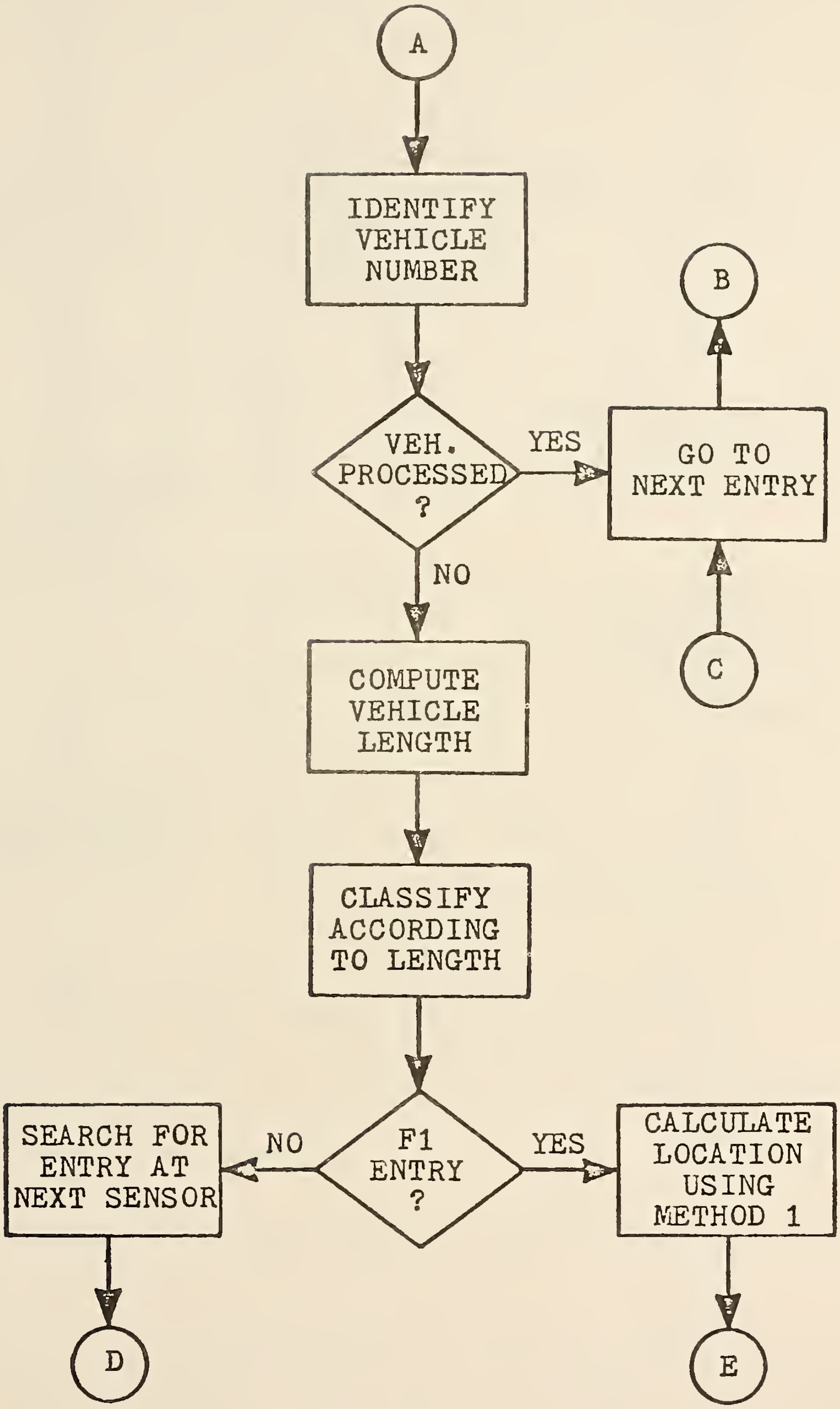


Figure 2. Flow Chart of the XVEH Subroutine (Cont'd)

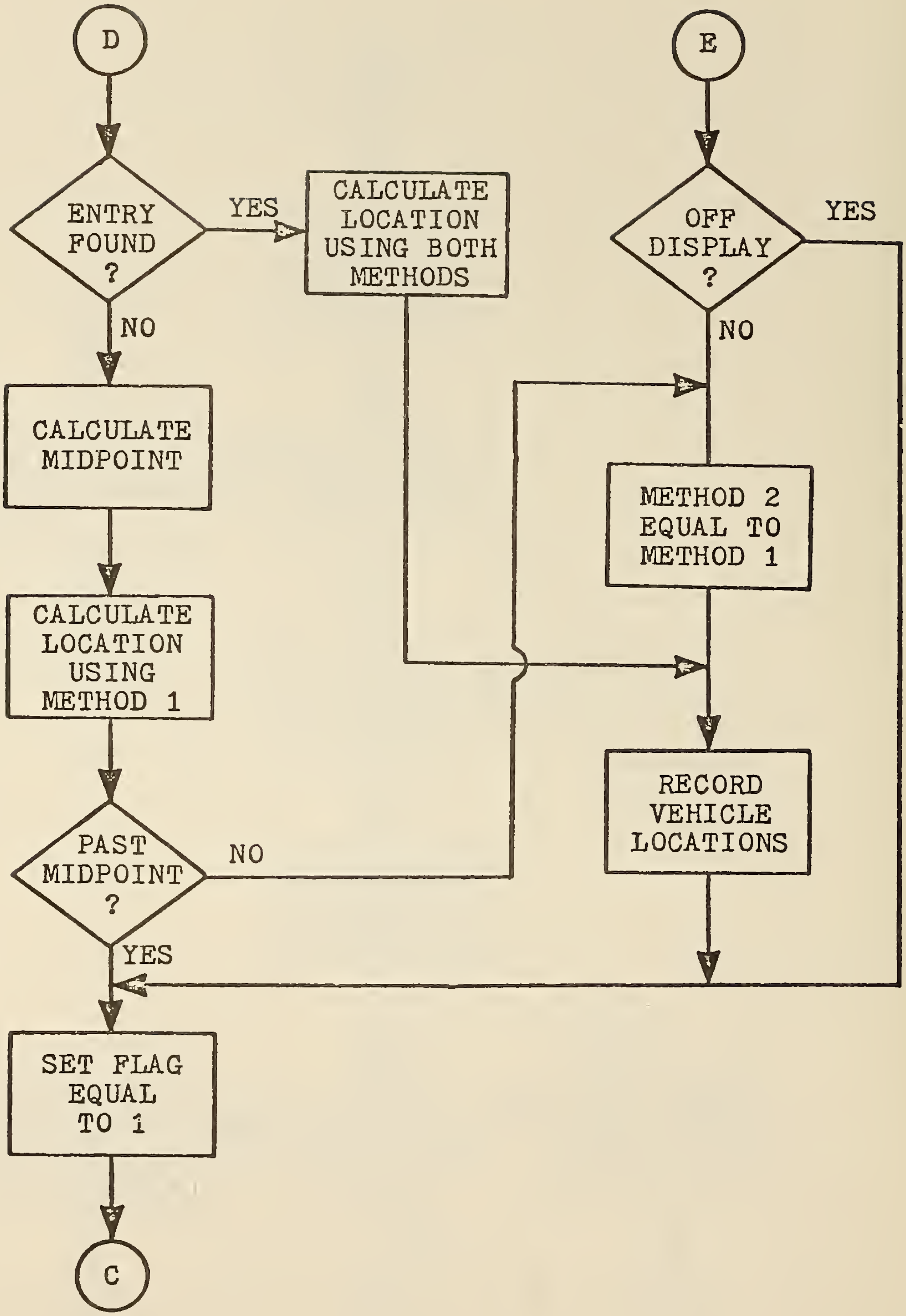


Figure 2. Flow Chart of the XVEH Subroutine. (Cont'd)

The routine sets an index variable to one, indicating sensor F1, and then tests if sensor lists for F1 through F4 have been examined. When all lists have been examined, the routine returns control to the main program; otherwise, another index variable is set equal to one to start the examination of the sensor activation entries. The first entry is examined to determine if it has occurred before or at the present simulation time. If the entry has not yet occurred, the routine increments the sensor index variable and begins to examine the next sensor's entries, provided all sensors have not be examined. If the entry has occurred, the vehicle number causing the entry is determined.

The vehicle flag is checked and if it is set, indicating the vehicle has been handled at a previous sensor, the routine advances to the next entry. If not set, the length of the vehicle, in feet, is calculated as the product of the measured vehicle velocity and its length in seconds. The vehicle is then classified according to its length as a truck, a car, or a motorcycle.

If the entry being examined is an F1 sensor entry, its location is calculated using method one. The measured velocity at the sensor is assumed to be continued through the sensor region, and the location of the vehicle is found by the following equation:

```
IXLOC = (RAWDAT (I,LCNT,3)) * (TIME - RAWDAT(I,LCNT,2)) + XSNS (I) + 0.5
```

The first term is the product of the vehicle velocity and the time since the entry, the second term is the location of the sensor, and the constant one-half is used for round off purposes. The result of using method one is the apparent "jerking" of the vehicle when its next sensor activation occurs.

If the vehicle has slowed down between sensors, the vehicle will appear to go beyond the next sensor location before the next sensor is activated. When the activation does occur, the vehicle will be jerked back to the sensor location. In the case of a vehicle increasing its speed between sensors the vehicle will jump forward when the next sensor activation occurs. The vehicle location is tested to determine if it is within the region displayed in the picture. If it is, the location is recorded according to vehicle type, the vehicle flag is set and the next entry is examined. If it is not, the vehicle flag is set without recording the location and the routine advances to the next entry.

If the entry had not been from sensor F1 the routine begins to search the sensor's entries for the entry caused by the vehicle under consideration. The search begins at the first list entry and continues to the end of the list if the proper entry is not found. If no entry is found, the vehicle is assumed to have cut out of the right lane and is plotted using method one until it reaches the midpoint of the sensor region. If an entry is found, the location is calculated using both methods.

Method two uses linear interpolation of the simulation time and the sensor entry times to determine the vehicle position. This method causes the vehicles to arrive at the next sensor at the proper time, and a smooth motion of the vehicles is achieved. The vehicle location is calculated using the following equation:

$$\mathrm{IXLOC} = \frac{\mathrm{TIME} - \mathrm{RAWDAT(I,LCNT,2)}}{(\mathrm{RAWDAT(I-1,NCNT,2)} - \mathrm{RAWDAT(I,LCNT,2)})} * 200$$

$$+ \mathrm{XSNS\ (I)} + 0.5$$

The first term is the linear interpolation term, and the second and third terms are the same as used in method one. The distance between sensors is 200 feet. This distance is multiplied by a fraction, less than one, which is determined by the amount of time since the sensor entry occurred.

After the vehicle location has been calculated using both methods, the results are recorded, the vehicle flag is set, and the routine advances to the next entry. When the lists for all four sensors have been examined, control is returned to the main program.

The second subroutine, FRMOUT, writes the required frame data into an on-line file. The flow chart for FRMOUT is shown is figure 3. FRMOUT is called immediately after the XVEH subroutine. The output data consist of the simulation time, the green band status, and the vehicle locations and is written such that each record is eighty characters in length. The data are subsequently used by the drawing program to draw the movie on a frame by frame basis at the CRT terminal. The next section describes the terminal hardware and the program used to draw the movie frames.

The Drawing Process

The drawing of each frame of the movie was accomplished with a Tektronix Model 4010 Graphics Terminal and the supportive Tektronix PLOT-10 Graphics Software. The 4010 is a storage tube CRT terminal with a typewriter keyboard for operator input. There are also operator controls for movement of the drawing cursor, but they were not used in this application of the terminal.

The PLOT-10 software is a set of subroutines available for use by the computer from the University of South Florida's Computer Library. The subroutines are user oriented and allow the drawing of pictures through programmer

Figure 3. Flow chart of the FRMOUT subroutine.

control of the movement of the CRT writing beam and its on-off status. When the beam is on, a line is drawn on the CRT face. When off, the beam is moved without drawing a line. Among the other subroutines available are routines which initialize the terminal, control switching to alphanumeric mode, and end a drawing session. Descriptions of the subroutines and examples of their use are given in the <u>Terminal Control System User's Manual</u>.

The drawing program consists of a main program and a series of subroutines. The main program allows the operator to control the drawing mode and to input variable values. The subroutines call the PLOT-10 routines to draw various portions of the movie frame. Figure 4 is a flow chart for the drawing program.

The main program initializes indexing variables and the terminal. The initialization of the terminal sets the transmission rate of the terminal in characters per second. It also erases the screen, causes entry into alphanumeric mode, sets margins, and carries out various graphics functions required before any drawing can be done. The program then instructs the operator to enter the drawing mode to be used.

The program allows two drawing modes: (1) movie mode and (2) search mode. If the operator selects the movie mode, he is prompted to enter a number which sets the number of frames to be drawn before another mode selection can be made. The program then requests the operator to enter the film code or name to be displayed in the movie frames and to enter the starting frame for the drawing session. The program reads from the frame data to the starting frame. When the starting frame is reached, the program

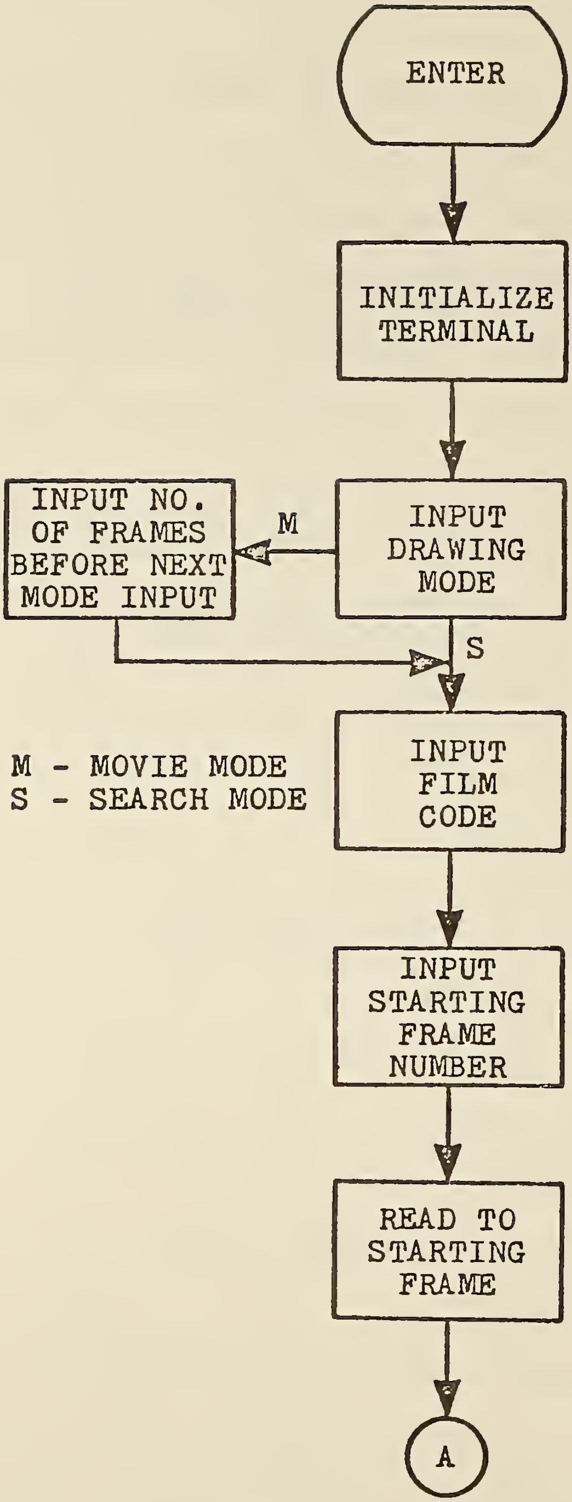


Figure 4. Flow chart of the drawing program.

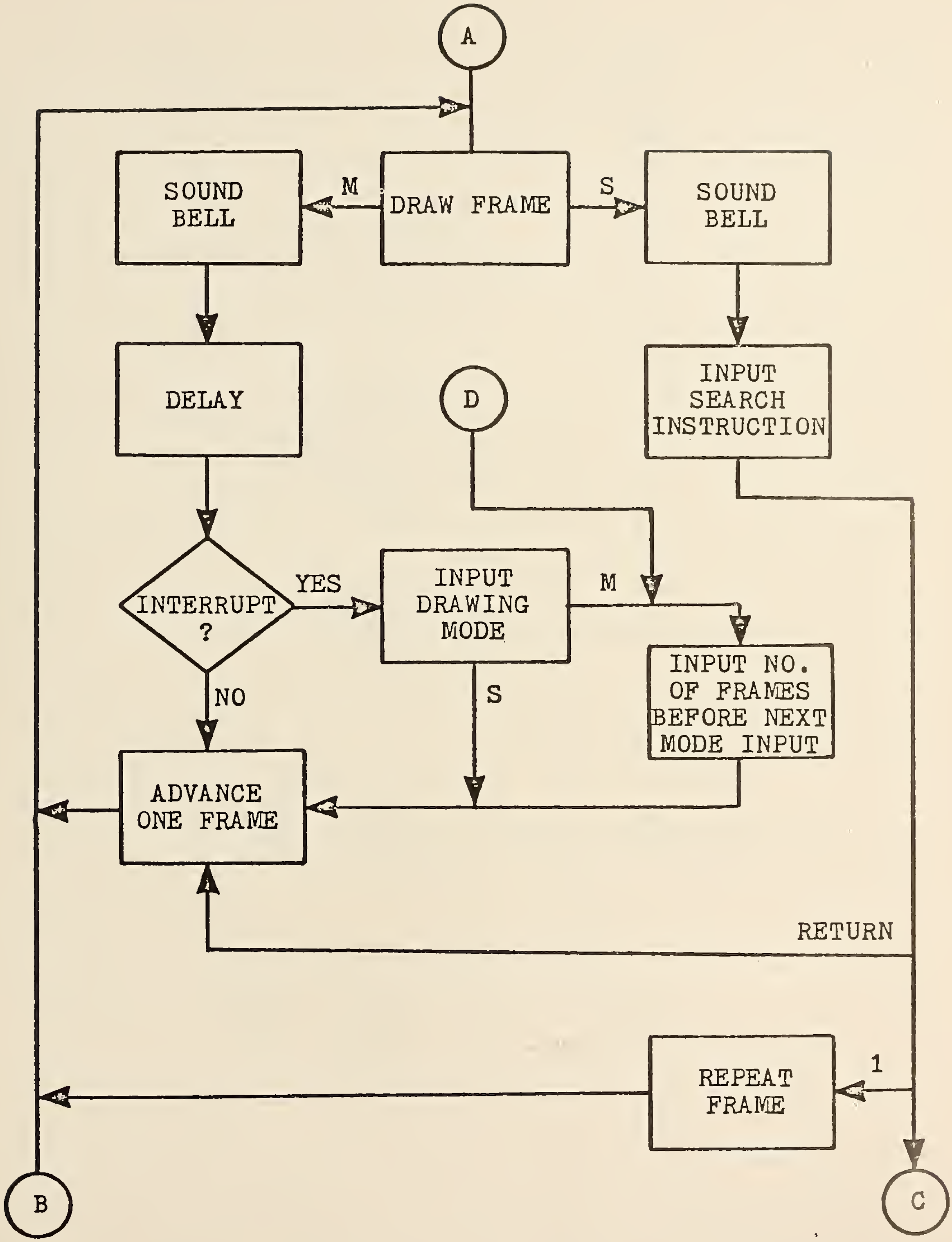


Figure 4 Flow Chart of the drawing program (Cont'd)

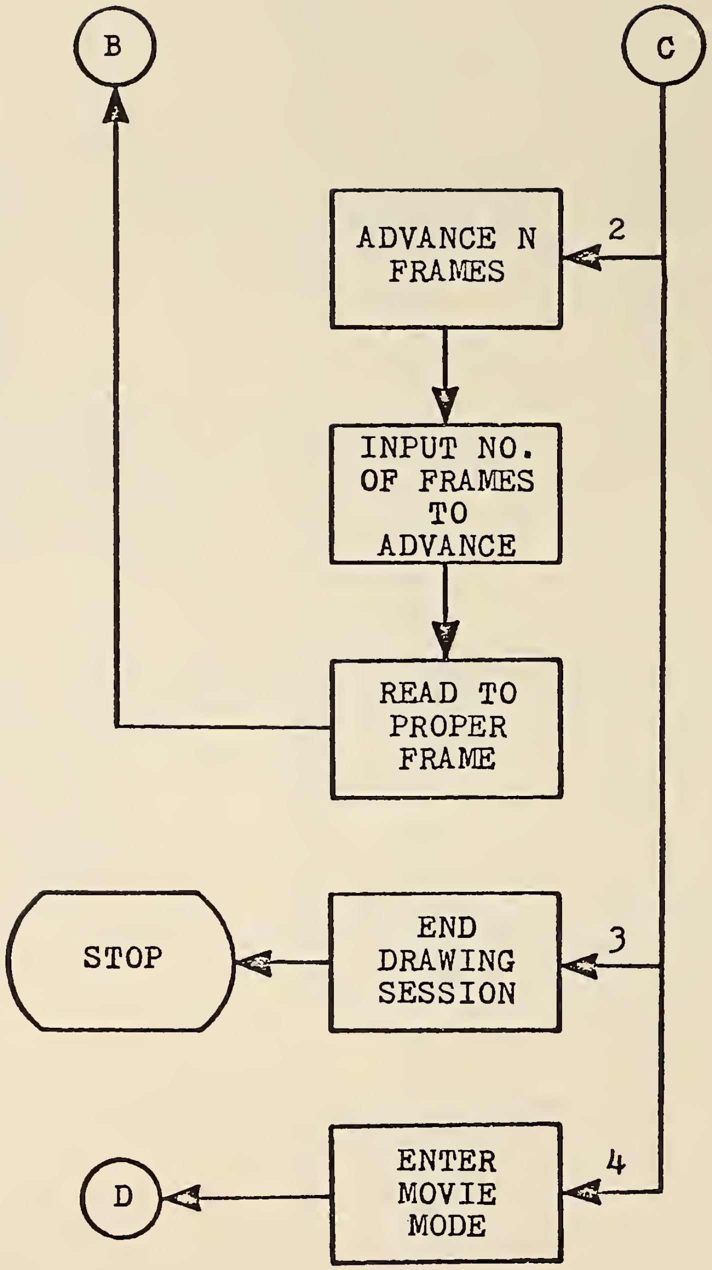


Figure 4 Flow Chart of the drawing program (Cont'd)

calls the subroutines which draw the picture and branches to another portion of the program determined by the drawing mode selected. In either case, the terminal bell, actually an electronic oscillator, is sounded to indicate the frame drawing is complete.

If the movie mode was selected, the program delays advancing to the next frame. The delay is achieved by causing short movements of the beam, in an off status, along the bottom of the display. The delay allows the frame to be photographed before the display is erased. The program then tests whether the number of frames required until the next mode selection have been drawn. If not, the program advances one frame and calls the drawing subroutines. If the number of frames selected have been drawn, the program interrupts to allow the re-entering of the drawing mode. The operator can again select which mode he wishes to use. If he selects the movie mode, the program will request the number of frames to be drawn before the next interrupt and the drawing process will resume.

The search mode of operation was used in the program development stage to provide a capability for manually stepping through the recorded data base without having to view each frame in a fixed sequence.

There are six subroutines used for drawing the various portions of the movie frame. Three of the subroutines draw the various types of vehicles. TRKDRW is called to draw vehicles which have been classified as trucks. CARDRW is called to draw cars, and CYLDRW is called to draw motorcycles. Each of the three subroutines calls a special subroutine which draws the wheels for the vehicles. The two remaining subroutines BCKRND and GRBDRW are used to draw the background and green band display respectively. The

BCKRND subroutine draws and labels the freeway, ramp, sensor, and merge point locations. It also writes the film code, the frame number, and the simulation time. The GRBDRW subroutine processes the green band data and in accordance to the band status, draws a line, representing the light being on, or simply moves over the light location without drawing a line. The subroutine also marks the ends of the green band display to facilitate subsequent data reduction when viewing the films.

Making the Movies

The physical equipment used to make the movies include the terminal, the camera, and the automatic shutter actuator. Figure 5 is a picture of the equipment set up. When making a movie, the terminal face plate is removed to increase the brightness of the display, and the camera is mounted on a sturdy tripod to insure camera stability during the period of filming.

The camera used was a Bolex H16. It is a spring-driven, 16 millimeter, motion picture camera with a single frame capability. Next day development capability at a local television film department prompted the selection of Kodak VNF 7240 color film. The exposure for single frame photography is controlled by the F-stop which was set at 1.1 for making the movies. A cable release and adapter were attached to the camera to reduce the chance of camera movement when the shutter was released. Any camera movement would cause the movie to jitter when shown at the normal frame rate. The cable release can be actuated by the operator or automatically.

Figure 6 is a block diagram of the automatic shutter release mechanism (ASRM). The major components of the ASRM are a sound switch, a silicon-controlled rectifier, a relay driver, a relay, a solenoid, and a

Figure 5 Equipment set up for computer generated green band movie production

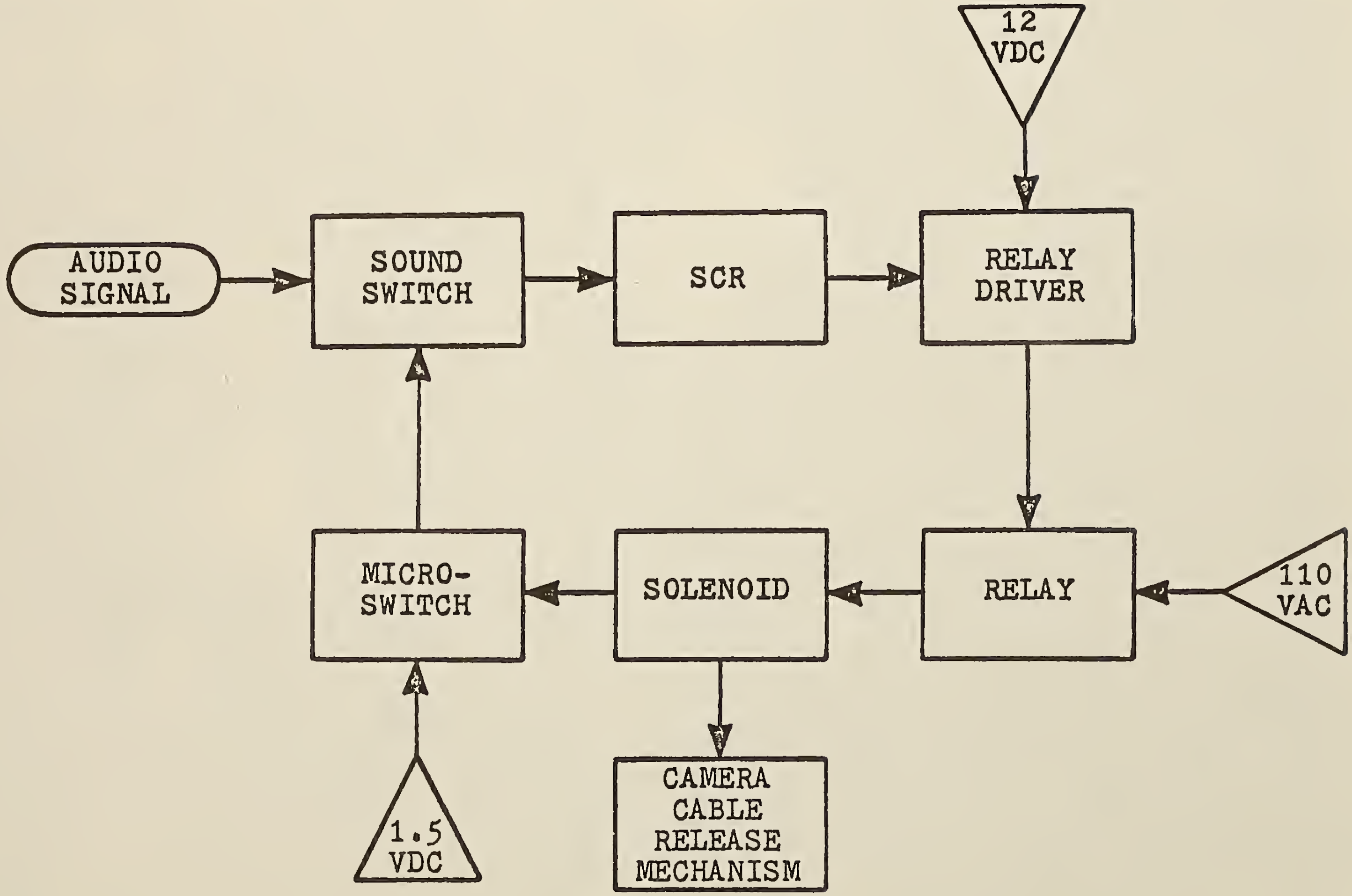


Figure 6 **Block diagram of the ASRM.**

power switch for the sound switch.

When the terminal bell sounds, the sound switch triggers the silicon-controlled rectifier (SCR). The SCR in turn triggers the relay driver which actuates the relay. The relay completes the circuit for the solenoid power, and the solenoid pushes the cable release to photograph the displayed movie frame. Once triggered, the sound switch remains on, and thus it was necessary to place a switch between the sound switch power supply and the sound switch itself. The switch turns off the sound switch power when the solenoid is actuated, thus closing the loop to provide automated operation without operator intervention.

The ASRM unit was constructed by interfacing the various components. The sound switch operates from a 1.5 VDC battery and when triggered provides a 0.7 volt output. Since this voltage is not adequate to drive a solenoid large enough to trigger the camera, it was used to trigger the SCR. The SCR is used as a switch to provide a ground condition for the relay driver. The relay driver circuitry was originally used to drive a small 12 VDC solenoid and was salvaged from an auto tape player. A 12 VDC relay was substituted for the solenoid. The relay provides the switching capability to control a solenoid operating on 110 VAC line voltage. The solenoid pushes the cable release and near the end of its stroke turns off the 1.5 VDC power to the sound switch by mechanical actuation of a microswitch. Turning off the sound switch causes the SCR, the relay driver, the relay, and finally the solenoid to turn off. The solenoid returns to its unactivated position which allows the microswitch to turn the sound switch back on. The unit is then ready for the next activation by the terminal

bell. The ASRM is also provided with a switch to disable the sound switch. This allows the system to be turned off while positioning adjustments are made and prevents undesired triggering of the unit.

After the operator has set up the equipment, he logs onto the terminal. He then types in the command "EXEC GBX". The GBX routine is a series of terminal commands that allocate the data sets required to execute the program and then calls the drawing program. The drawing program itself is stored in the form of a load module which has already been compiled. When this module is called, the execution of the drawing program described in the previous section begins.

The operator enters the search mode and draws the first frame. The camera is then focused and the starting frame is photographed manually. The operator can then turn on the ASRM and instruct the drawing program to enter the movie mode. After inputing the number of frames to be drawn before the next mode interrupt, the program begins to automatically draw and photograph the movie frames.

The exposed film is developed and, when played back at 24 frames per second, shows the dynamic movement of the vehicles and the green bands on a real time basis as generated by the Green Band Control Program Simulation.

Analysis of Movie Data

Recall that two methods are used to calculate the vehicle locations. The first method depicts the effects of using the measured velocity between the sensors and results in an apparent jerking motion of the vehicles. The second method uses linear interpolation to achieve smooth vehicular motion.

Figures 7 through 9 show the effects of the two methods of calculating the vehicle location. Each vehicle along the freeway is displayed twice in the frame. The top vehicle shows the location of the vehicle as calculated by linear interpolation and moves smoothly along the freeway. The bottom vehicle is the vehicle which appears to jerk.

Figure 7 shows two vehicles (with two representations of each vehicle) approaching the sensor F3 location. Note that the top representation of both vehicles is ahead of the corresponding bottom vehicle. Since both top vehicles are pulling ahead, they must have increased their velocities after crossing over sensor F4. The top vehicles will arrive at sensor F3 when the actual sensor entry occurred, while vehicles, whose positions are projected ahead by maintaining the sensor F4 velocities, will be jerked forward to the sensor location. Figure 8 shows the top vehicle about to cross the sensor. The next frame, Figure 9, shows that the bottom vehicle has jumped forward to the sensor.

The crossing of sensor F3 by the lead vehicle has given the Green Band Control Program Simulation an update for the ETA of the vehicle. Since the vehicle velocity has increased, its ETA has decreased as the vehicle will arrive at the merge point sooner. The change in ETA has caused the trailing edge of the first band to move forward along the display, as can be seen in Figures 8 and 9. The band edge movement occurred within a period of 42 milliseconds and is an example of an undesired instability occurring in the operation of the on-line system.

Several movies, using various simulated band control strategies, were made in the manner presented in this section. A detailed analysis of the movies was carried out and a finalized band control strategy was selected for implementation in the on-line system at the August, 1976, TAC meeting.

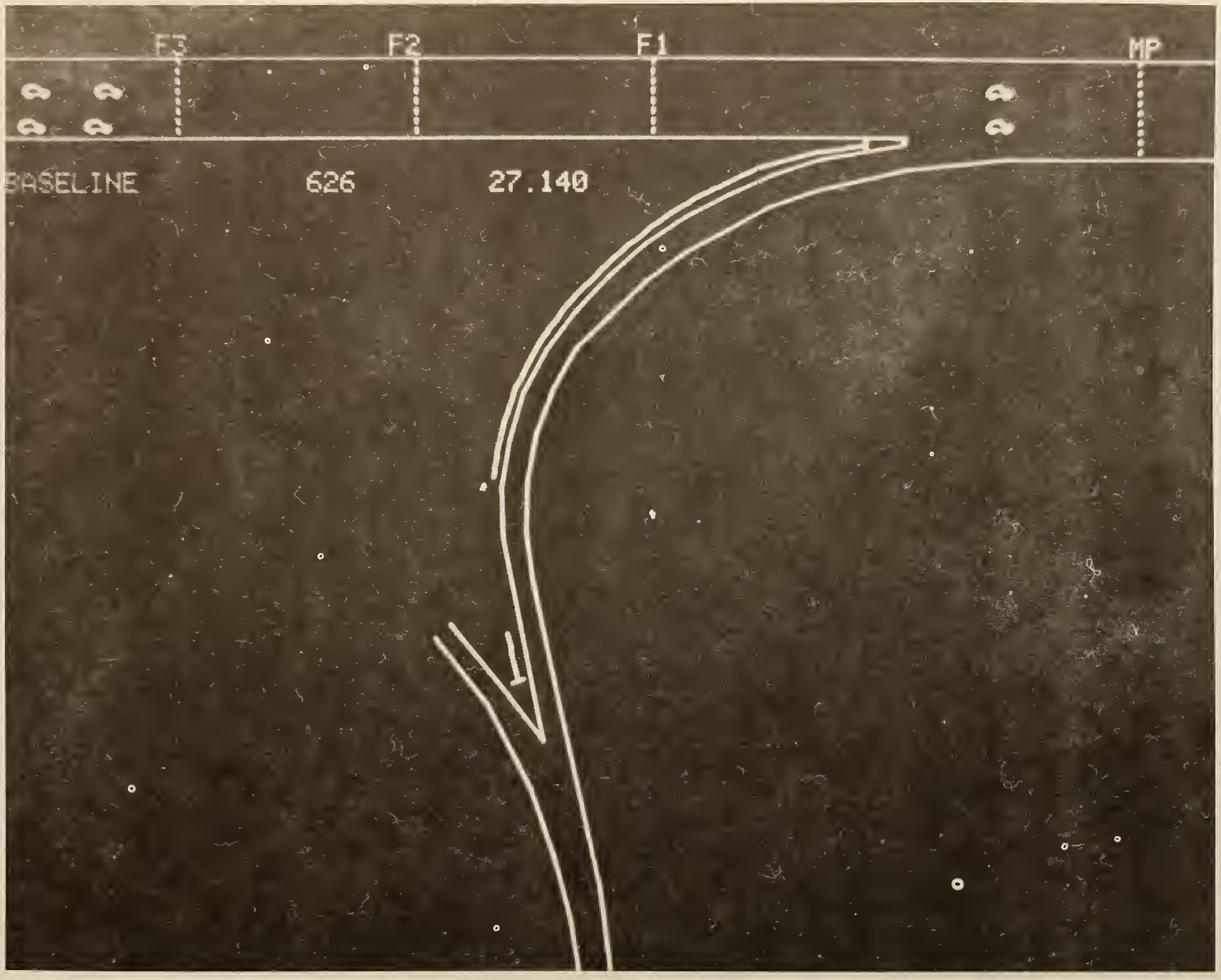


Figure 7. Frame 626 of a green band movie.

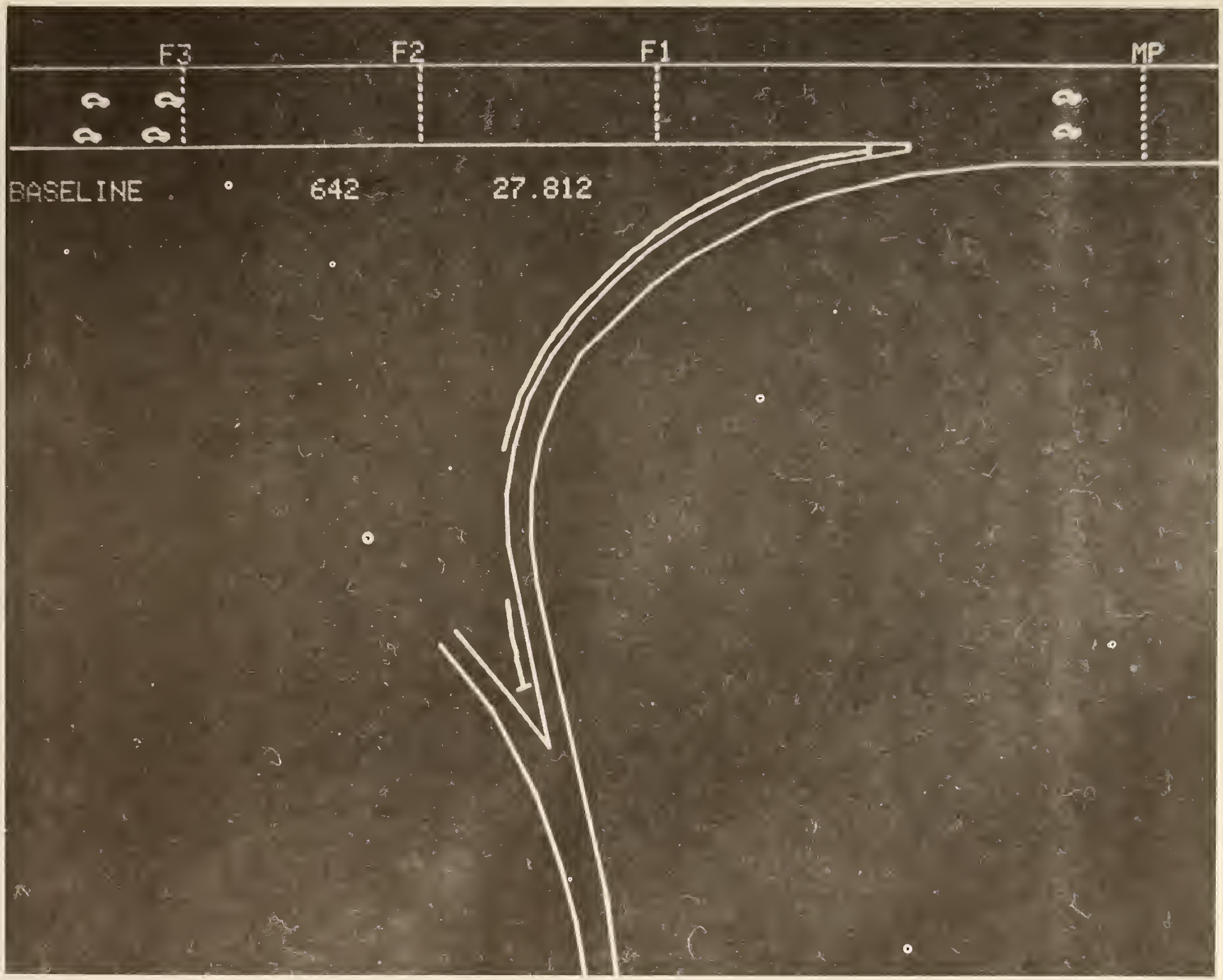


Figure 8. Frame 642 of a green band movie.

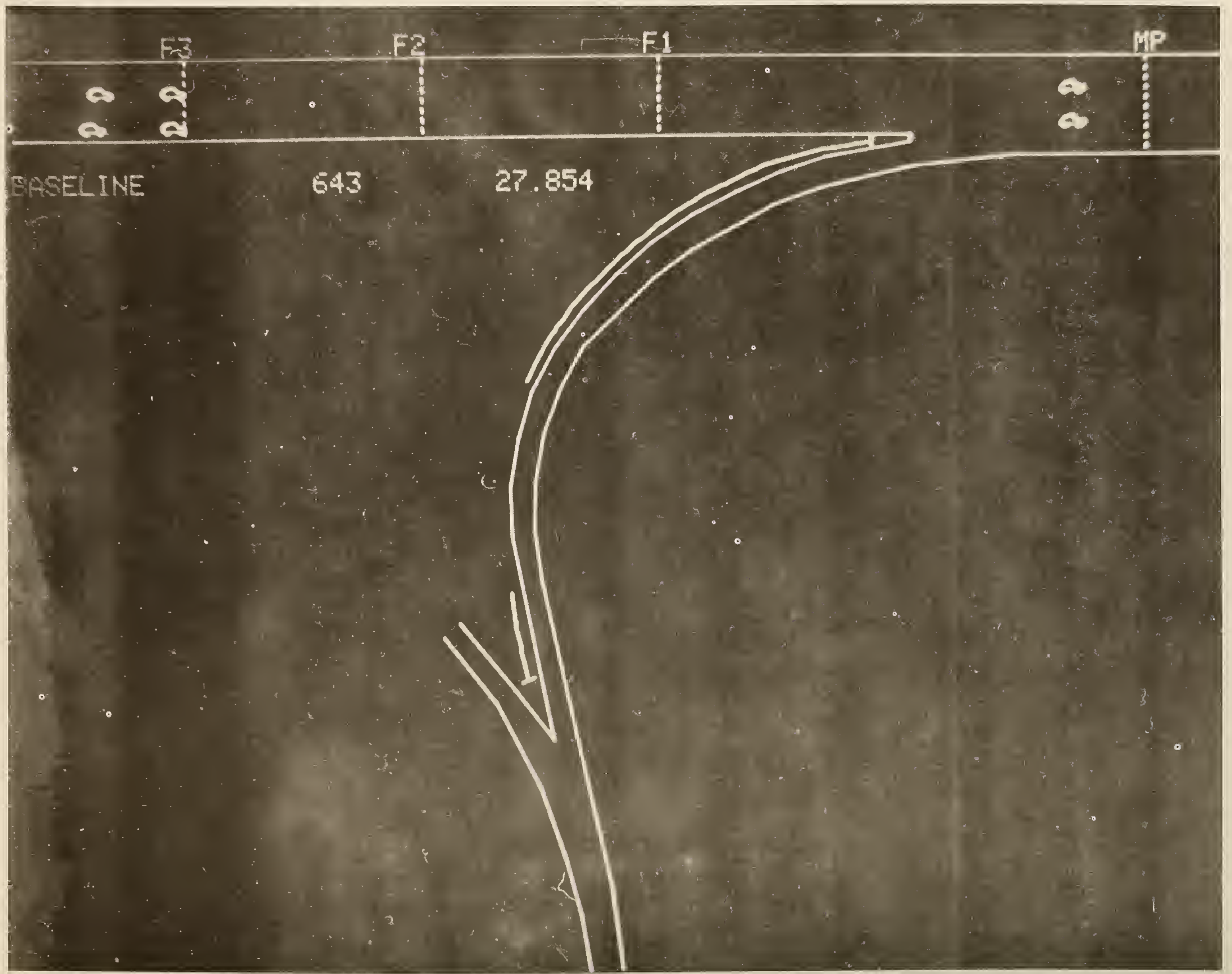


Figure 9. Frame 643 of a green band movie.

The movies have been invaluable in visually correlating the unstable band edge movements to the vehicle actions along the freeway. In the first film produced, it was noted that the trailing edge of the bands would allow a ramp vehicle to merge very close to a freeway vehicle. A later simulation run showed the generation of a band whose trailing edge overlapped the leading edge of the next band. An investigation of the simulation found a sign change, carried out by the on-line system, was not accounted for in the simulation. The incorrect sign caused the trailing edge of all bands to be displaced toward the beginning of the display. Therefore, the films have not only proven to be an aid in analysis of the band edge instability, but have contributed to the validation of the simulation's representation of the on-line system.

The requirement to redraw the entire picture for each frame, imposed by the use of a storage tube CRT, greatly increases the time required to produce a movie. With proper syachronization a refreshing CRT terminal would allow the photographing of the simulation in real time. With the equipment used on this project, it required eight to nine hours of terminal/operator time to produce a thirty second movie.

SECTION III: FINAL BAND SMOOTH ALGORITHM

In order for an algorithm to reduce or smooth the band edge instabilities, it must first identify the area of instability, obtain some measure of the size of the edge movement, and classify the change as a particular type of movement.

The identification of the edge movement is obtained through the comparison of the current display status with the new directed output from the normal GBS and GBUD processing. The comparison takes place by adding the individual light statuses, 0 or 1 in the on-line system, and looking at the result. Where both values were 0's the sum will be a 0, and where both were 1's the sum will be 2. But, where the values are unlike the result will be a 1 and the 1 will indicate a change has taken place.

For each light status change a 1 will be generated. Therefore, the number of 1's in a series will give a measure of the size of the band change.

The classification of the type of change can be made by looking at the values just before and after the 1's. For example, if the value just before the 1's was a 2 and the value after was a 0, we are dealing with a change in a leading band edge. Figure 10 illustrates this.

Current display	111111100000	
New command	111111110000	
Sum	222222210000	Change in band edge

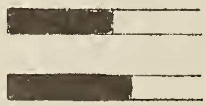


FIG. 10

In the example only one 1 was generated by the change and in a forward direction. This is a normal advance of a leading band edge. So one can classify the type of change by the values before and after the 1's. The example would then be a 2,0 type change. If the 1 had occurred at either end of the display a value of -1 would be assigned to the appropriate before or after value.

In classifying the band movement, the algorithm attempts to smooth the change and reduce rapid adjustment of the band edges. The nine possible combinations for the before and after values are listed below.

0,0	2,0	-1,0
0,2	2,2	-1,2
2,-1	2,-1	-1,-1

In addition to the classifications above the algorithm must determine if the band edge change was either forward or backward along the display. This is accomplished by testing the value of the current display at the location of the 1 or, in the case of greater than one 1, the first 1 of the series. Depending on the particular type of edge (leading or trailing) this value will give the direction of the change. For a leading edge Fig.III.1, a 0 would indicate a forward movement. The values for a trailing edge would be the opposite of the leading edge for the same movement.

At present when a ETA update causes an adjustment in the band edge, the edge is moved immediately and takes place in one output cycle. With an output every .01 seconds the band edge normally moves between .44 and .66 feet per cycle. If the output cycle was lengthened, the movement between cycles would be increased.

Cycle time	Movement: @ 44 ft/sec.	@66ft/sec.
.02	.88ft	1.32ft
.03	1.32ft	1.98ft
.04	1.76ft	2.64ft
.05	2.20ft	3.30ft
.06	2.64ft	3.96ft
.07	3.08ft	4.62ft

It can be seen that above .06 seconds the band edge at 66ft/sec advances greater than one light. Therefore, if smooth movement under normal progression is desired, the maximum time between outputs would be .06 seconds.

When it has been determined by the algorithm that an adjustment is necessary, and does not involve a possible cut in, the edge is adjusted one light (4ft) per output cycle. This leads to adjustment speeds of the band edges at various output cycle times as shown below.

	Edge Adjustment Speed	
Cycle time	ft/sec	mph
.02	200.00	136.36
.03	133.33	90.91
.04	100.00	68.18
.05	80.00	54.54
.06	66.67	45.46

The table shows that the band edge adjustment speed can be changed by changing the time between outputs. If a .06 second cycle time were selected the band edge adjustments would take place at slightly higher than the merge speed. However, since the normal edge can be moving at 66ft/sec the actual adjustment of an edge may take considerable time. For example if the band edge moved forward four lights or 16ft and was in the 45 mph segment this adjustment would be completed in $\frac{16}{66.67-66}$ or 23.88 seconds which is unacceptable. Times for adjustment of the same 16 feet adjustment at the various cycles times are given below.

Cycle time	Time to adjust 16ft (sec.) @ 66ft/sec	@ 44ft/sec	
.02	0.12	0.10	
.03	0.24	0.18	
.04	0.47	0.29	
.05	1.14	0.44	(Implemented in Revised Tampa System)
.06	23.88	0.71	

The algorithm has built in safeguard for vehicle cut ins. If a vehicle cuts into an all green band segment the control program will generate a break in the band. The minimum break would consist of a leading and trailing headway and the length of the vehicle. These headway values are 1.2 and 0.4 seconds, respectively. Therefore a minimum (neglecting the vehicle length) break would be 1.6 seconds. At 44ft/sec and 66ft/sec it works out to be 70.4 to 105 feet. On the display this would be 17 to 26 lights. Therefore within the algorithm, when 15 or more lights are extinguished it is assumed to be a vehicle cut-in. Also special

consideration was given to cases occurring at the ends of the display where a cut-in may not extinguish 15 lights. Here the algorithm assumes a cut-in if 2 or more lights are extinguished. When a possible cut-in occurs the display is immediately changed to reflect the occurrence.

A desired feature of the system is that no bands be generated other than at the start of the display. The generation of a band in the midst of extinguished lights would cause a series of 1s to appear between a leading and following 0. Also the commanded destruction of a band in the event of a gap closure would cause this combination of leading and trailing values. In either case the algorithm causes extinction of the lights and thus shows immediately any gap closures (complete band destruction) and eliminates generation of the undesired new bands.

The attached flow chart for the algorithm shows the possible cases and the measures taken for each.

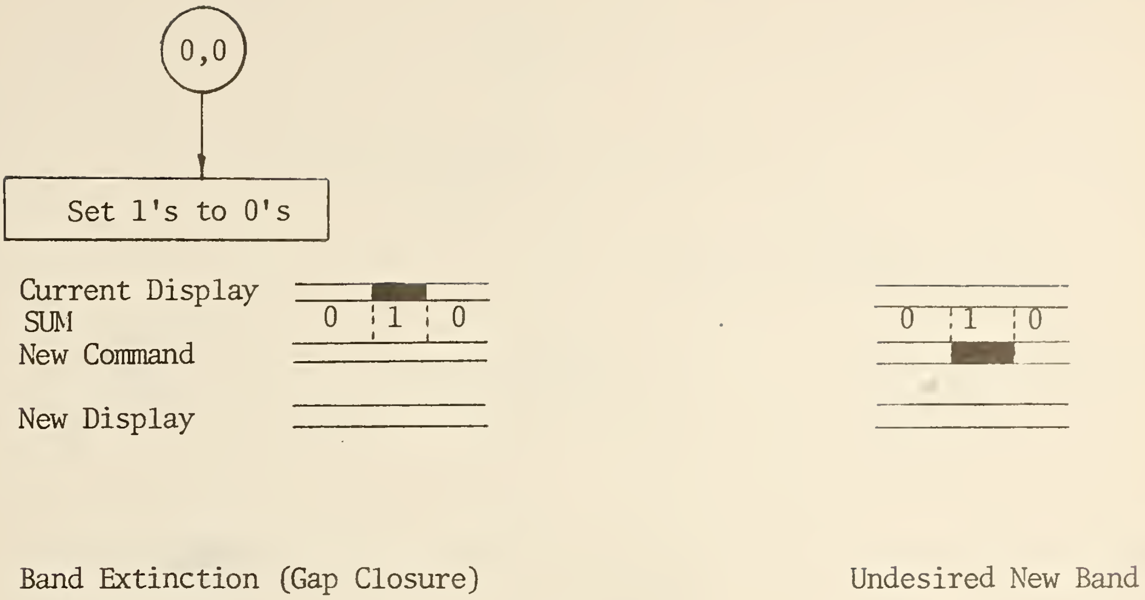


Figure 2 - Band Creation or Extinction Along the Display

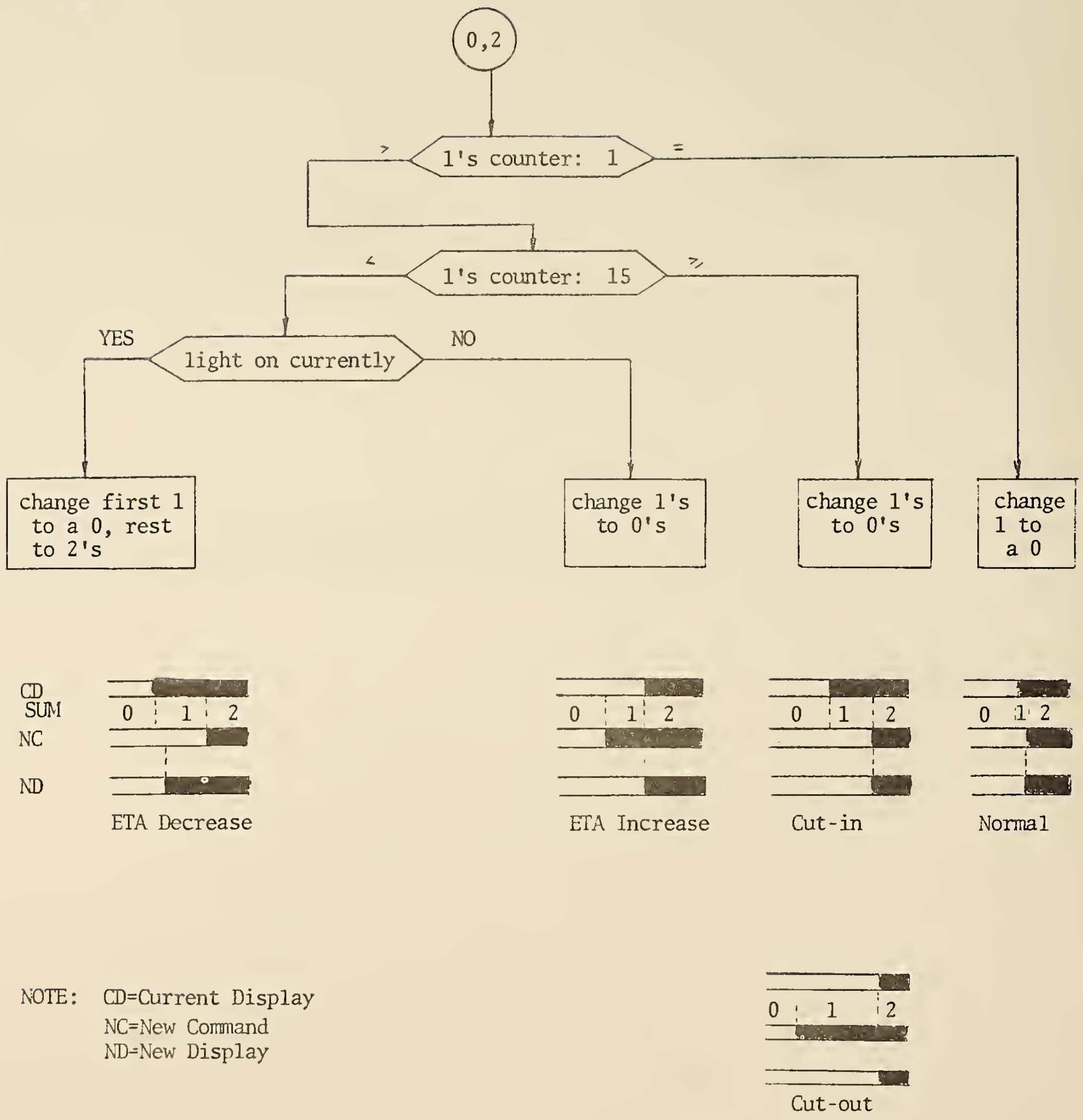
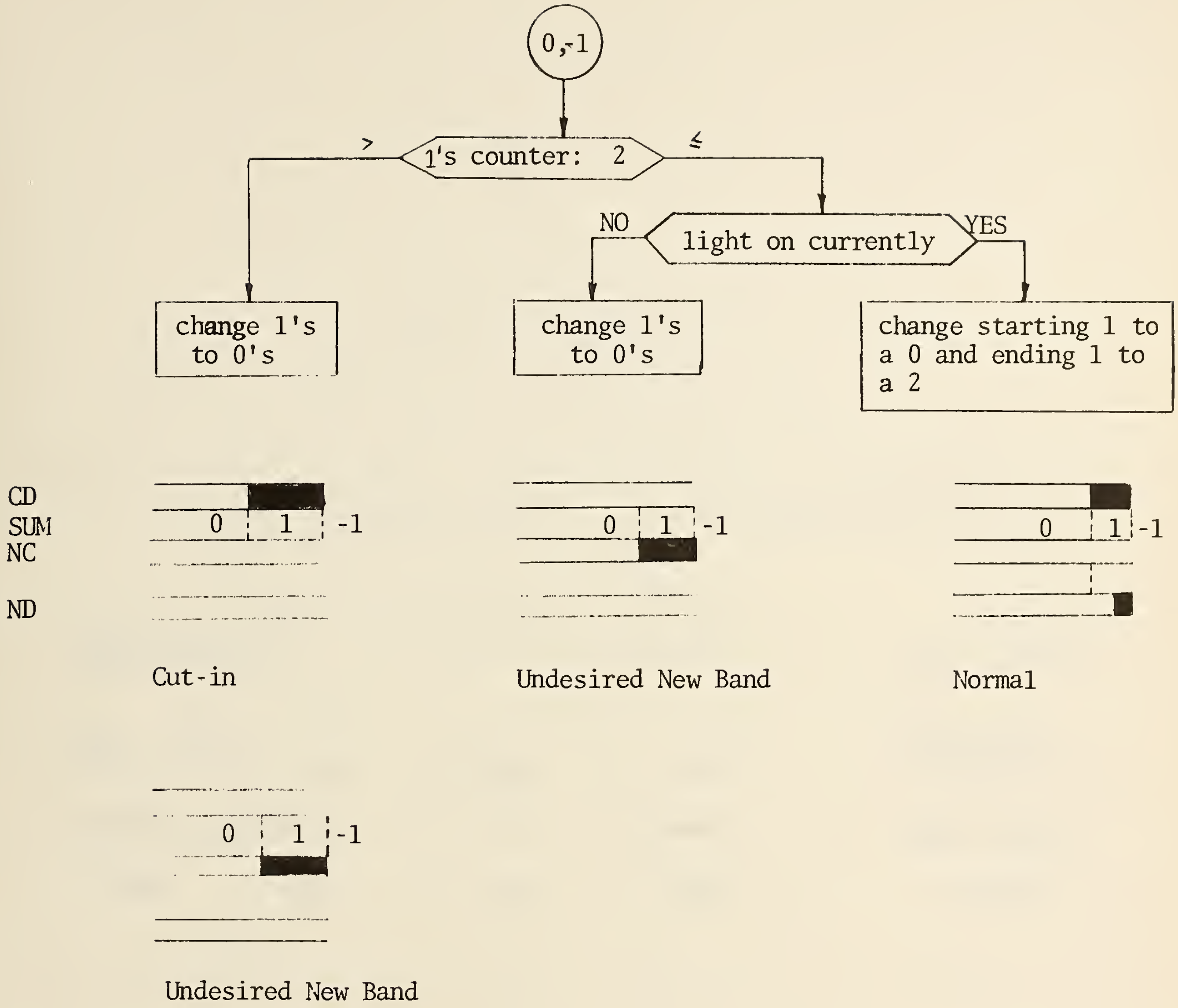


Figure 12 - Trailing Edge Variations Along the Display



NOTE: CD = Current Display
NC = New Command
ND = New Display

Figure 13 - Trailing Edge Variations at End of Display

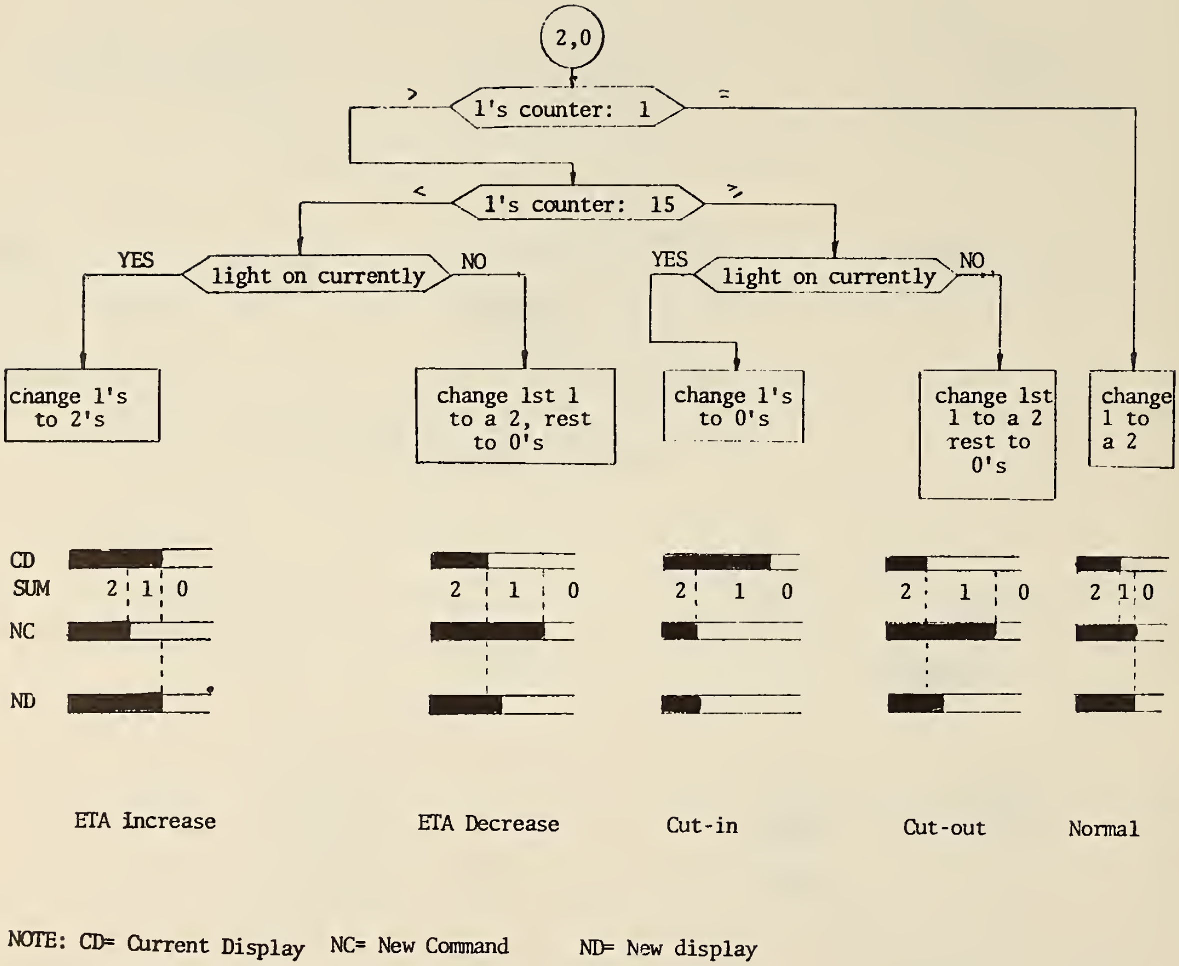
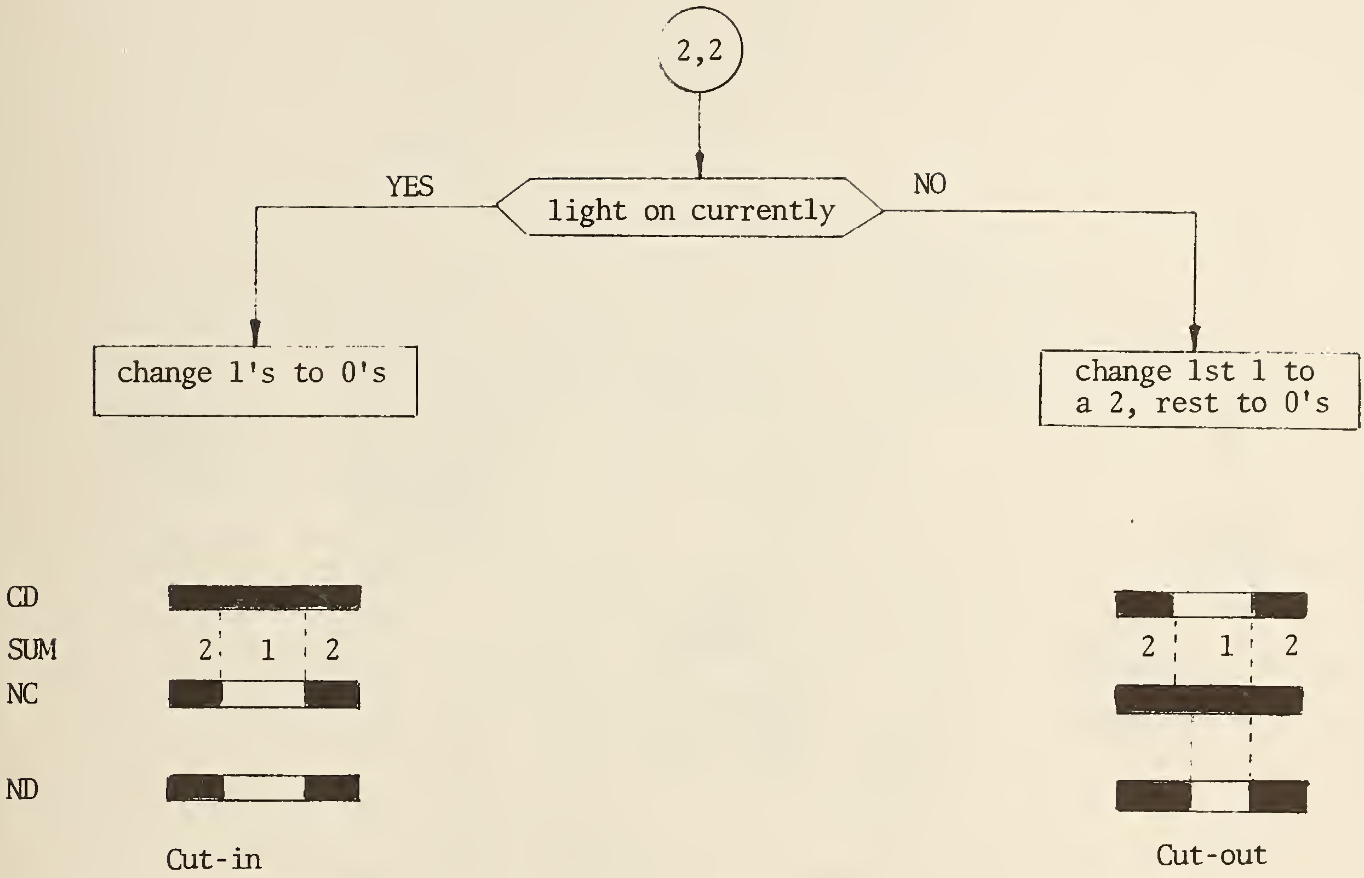
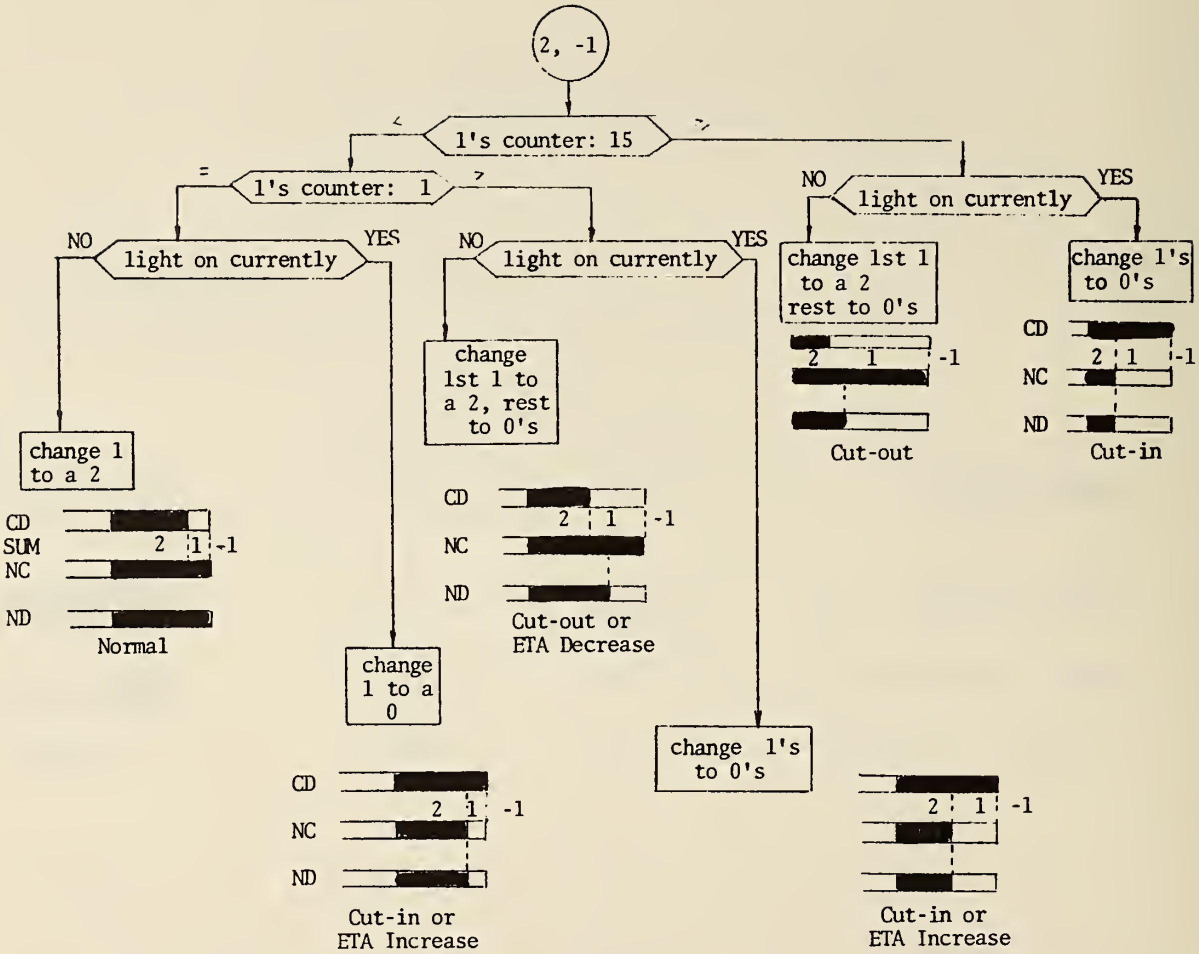


Figure 14 - Leading Edge Variations Along the Display



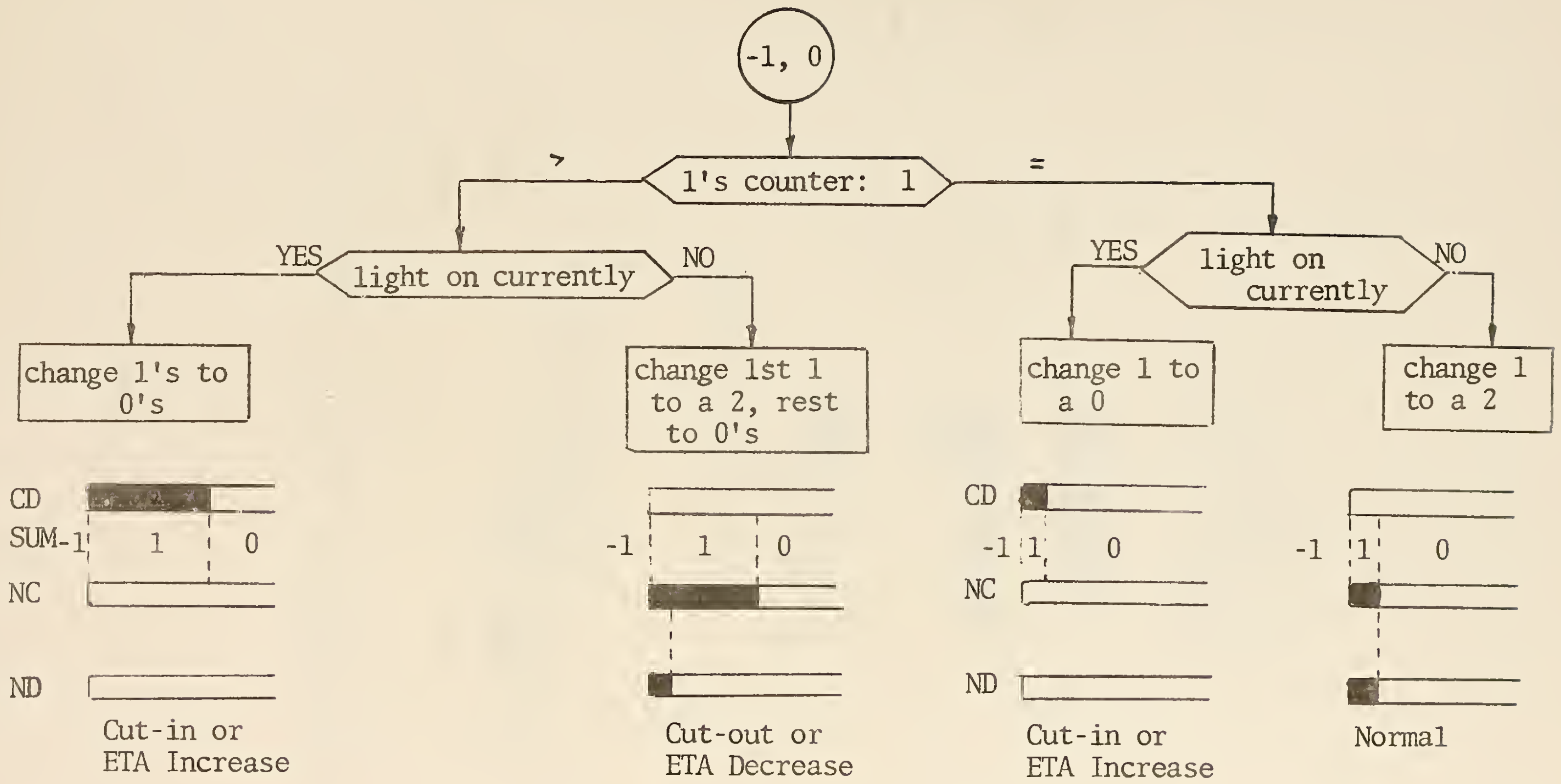
NOTE: CD= Current Display
NC= New Command
ND= New Display

Figure 15 - Band Segmentation Along the Display



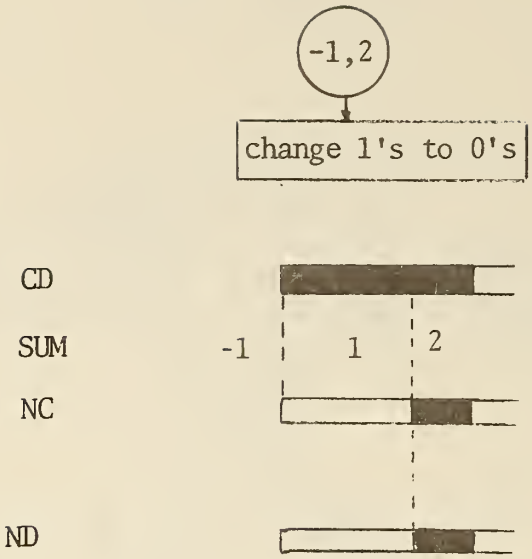
NOTE: CD= Current Display NC= New Command ND = New Display

Figure 16 - Leading Edge Variations at End of Display



NOTE: CD = Current Display
NC = New Command
NC = New Display

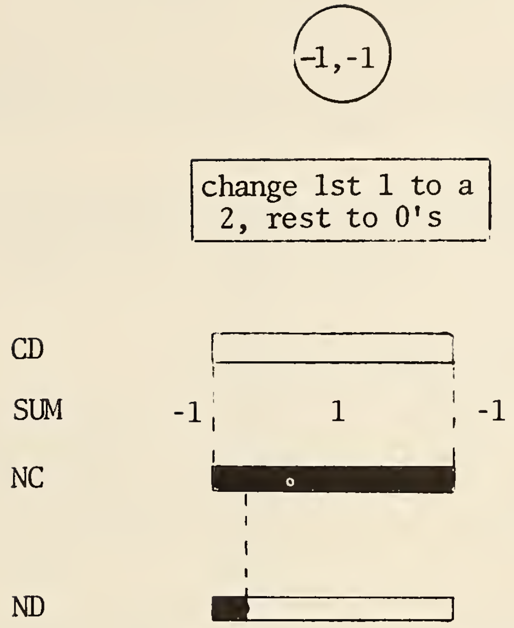
Figure 17 - Leading Edge Variations at Start of Display



Cut-in or ETA Decrease

NOTE: CD = Current Display
NC = New Command
ND - New Display

Figure 18 - Trailing Edge Variations at Start of Display



Initialization

NOTE: CD = Current Display
NC = New Command
ND = New Display

Figure 19 - Complete Display Variations

REFERENCES

1. Bauer, C.S., "Operating Strategies and Related Performance Evaluations for a Moving Merge Control System," Doctoral Dissertation, University of Florida, March, 1975.

2. Implementation and Evaluation of a Moving Merge Control System in Tampa, Report to the Florida Department of Transportation, University of Florida Transportation Research Center, July, 1975:

    Volume I : System Description
    Volume II : Program Documentation
    Volume IIa: Program Listings
    Volume III: Analysis of Vehicle Presence Detector Data
    Volume IV : Analysis, Conclusions, Recommendations

3. Newman, W. M., and Sproull, R. F. 1973. Principles of interactive computer graphics. New York: McGraw-Hill.

4. Rodgers, D. F., and Adams, J. A. 1976. Mathematical elements for computer graphics. New York: McGraw-Hill.

5. Terminal control system user's manual. 1972. Beaverton, Oregon: Tektronix Inc.

APPENDIX A

FORTRAN LISTINGS OF THE DRAWING PROGRAM AND THE GREEN BAND CONTROL PROGRAM SIMULATION

```
***********************************
*                                 *
*                                 *
*  LISTING OF THE DRAWING PROGRAM *
*                                 *
*                                 *
***********************************

      COMMON/NSTRT/ LSTART
      DIMENSION IXGR(153),IYGR(153),IXRL(20),IYRL(20),IFIELD(152),
     *IXRR(16),IYRR(16),NTKLOC(20),NCRLOC(20),NCYLOC(20)
      INTEGER FILMNM(2)
      LSTART = 1
      NFOP = 0
C
C     INITIALIZE THE TERMINAL.
      CALL INITT(30)
C
C     READ THE DRAWING MODE.
      WRITE(6,10)
   10 FORMAT (1X,'THE MOVIE MODE DRAWS N FRAMES BETWEEN OPERATOR INPUTS.
     *'/1X,'THE PROGRAM CAN BE STOPPED WHILE IN THE MOVIE MODE BY THE BR
     *EAK KEY.'/1X,'TYPING END, AND THEN HIT RETURN.'/1X,'IF THE MOVIE M
     *ODE IS TO BE ENTERED NOW ENTER A  1, IF NOT HIT RETURN.')
      READ (5,20) IOP
   20 FORMAT (1I1)
      IF (IOP.NE.1) IOP = 2
C
C     READ THE VALUE OF N, IF APPLICABLE.
      IF (IOP.EQ.2) GO TO 50
      WRITE (6,30)
   30 FORMAT (1X,'ENTER THE NUMBER OF FRAMES TO BE DRAWN BEFORE THE NEXT
     *OPERATOR INPUT, THE FORMAT IS 1I4')
      READ (5,40) NN
   40 FORMAT (1I4)
```

```
C
C     READ THE FILM CODE.
   50 WRITE (6,60)
   60 FORMAT (1X,'ENTER THE FILM CODE, THE FORMAT IS 2A4.')
      READ (5,70) FILMNM
   70 FORMAT (2A4)
C
C     READ THE STARTING FRAME NUMBER.
      WRITE (6,80)
   80 FORMAT (1X,'ENTER THE STARTING FRAME NUMBER, THE FORMAT IS 1I4.')
      READ (5,40) NFRAME
C
C     READ TO THE STARTING FRAME FROM THE FRAME DATA.
      NFR = NFRAME
   90 DO 130 NF = 1,NFR
      READ (3,100) TIME,(IFIELD(I),I = 1,72)
  100 FORMAT (1F9.3,72I1)
      READ (3,110) (IFIELD(I),I = 73,152)
  110 FORMAT (1X,80I1)
      READ (3,120) (NTKLOC(I),I = 1,7),(NCRLOC(I),I = 1,10),(NCYLOC(I),I
     *= 1,3)
  120 FORMAT (1X,20I4)
      READ (3,120) (NTKLOC(I),I = 11,17),(NCRLOC(I),I = 11,20),(NCYLOC(I
     *),I = 11,13)
  130 CONTINUE
C
C     DRAW THE FRAME.
  140 CALL ERASE
      CALL BCKRND(FILMNM,TIME,NFRAME)
      CALL GRBDRW(IFIELD)
  150 LEND = LSTART + 6
      DO 160 NTRK = LSTART,LEND
      IF (NTKLOC(NTRK).EQ.9999) GO TO 170
      CALL TRKDRW(NTKLOC(NTRK))
  160 CONTINUE
  170 LEND = LSTART + 9
      DO 180 NCAR = LSTART,LEND
      IF (NCRLOC(NCAR).EQ.9999) GO TO 190
      CALL CARDRW(NCRLOC(NCAR))
  180 CONTINUE
```

```
  190 LEND = LSTART + 2
      DO 200 NCYL = LSTART,LEND
      IF (NCYLOC(NCYL).EQ.9999) GO TO 210
      CALL CYLDRW(NCYLOC(NCYL))
  200 CONTINUE
  210 IF (LSTART.EQ.11) GO TO 220
      LSTART = 11
      GO TO 150
  220 LSTART = 1
C
C     GO TO THE PROPER MODE SEGMENT OF THE PROGRAM.
      GO TO (230,280), IOP
C
C     THE MOVIE MODE SECTION OF THE PROGRAM ALLOWS THE DRAWING OF N
C     FRAMES WITHOUT OPERATOR INTERVENTION.  IT CAN BE STOPPED BY THE
C     BREAK KEY, TYPING END, AND THEN HIT THE RETURN KEY.
C
C     INCREMENT THE FRAMES SINCE LAST OPERATOR INPUT COUNTER.
  230 NFOP = NFOP + 1
C
C     SOUND THE TERMINAL BELL.
      CALL BELL
      CALL BELL
C
C     DELAY THE PROGRAM TO ALLOW THE PICTURE TO BE TAKEN.
      CALL MOVABS(0,0)
      DO 240 MOVIT = 1,100,4
  240 CALL MOVABS (MOVIT,0)
C
C     INTERRUPT IF N FRAMES HAVE BEEN DRAWN.
      IF (NFOP.EQ.NN) GO TO 260
C
C     INCREMENT THE FRAME COUNTER AND ADVANCE TO NEXT FRAME.
  250 NFRAME = NFRAME + 1
      NFR = 1
      GO TO 90
C
C     GO THROUGH MOVIE OPTION INTERRUPT.
  260 CALL ANMODE
      CALL ERASE
```

```
      CALL HOME
      WRITE (6,270) NFOP
  270 FORMAT (1X,1I4,2X,'FRAMES HAVE BEEN DRAWN SINCE THE LAST OPERATOR
     * INPUT.'/1X,'IF YOU WISH TO CONTINUE THE MOVIE MODE ENTER A  1, IF
     * NOT HIT RETURN.')
      READ(5,20) IOP
      IF (IOP.NE.1) IOP = 2
      NFOP = 0
  275 IF (IOP.EQ.1) WRITE (6,30)
      IF (IOP.EQ.1) READ (5,40) NN
      GO TO 250
C     THE SEARCH MODE IS THE DEFAULT IF THE MOVIE MODE IS NOT
C     SELECTED.  IT ALLOWS THE OPERATOR TO REPEAT A FRAME, DRAW
C     THE NEXT FRAME, ADVANCE MORE THAN ONE FRAME, OR END THE SESSION.
C
C     AN OPERATOR INPUT IS REQUIRED AFTER EACH FRAME IS COMPLETED.  THE
C     INPUT DETERMINES WHAT IS TO BE DONE NEXT.  THE CODES ARE:
C
C             INPUT       RESULT
C             RETURN      NEXT FRAME IS DRAWN
C               1         FRAME IS REPEATED
C               2         ADVANCE ONE OR MORE FRAMES
C               3         END THE SESSION
C               4         ENTER THE MOVIE MODE
C
C     SOUND THE TERMINAL BELL.
  280 CALL BELL
      CALL BELL
      CALL HOME
      CALL ANMODE
C
C     WITHOUT ERASING THE PICTURE, READ THE SEARCH INSTRUCTION.
      READ (5,20) INST
C
C     CARRY OUT THE INSTRUCTION.
      IF (INST.EQ.1) GO TO 290
      IF (INST.EQ.2) GO TO 300
      IF (INST.EQ.3) GO TO 320
      IF (INST.EQ.4) GO TO 340
      GO TO 250
```

```
C
C     REPEAT THE FRAME.
  290 GO TO 140
C
C     ADVANCE ONE OR MORE FRAMES.
  300 CALL ERASE
      WRITE (6,310)
  310 FORMAT (1X,'ENTER THE NUMBER OF FRAMES TO ADVANCE, THE FORMAT IS 1
     *I4.')
      READ (5,40) NFADV
      NFRAME = NFRAME + NFADV
      NFR = NFADV
      GO TO 90
C
C     END THE DRAWING SESSION.
  320 CALL ERASE
      WRITE (6,330)
  330 FORMAT (1X,'THE DRAWING SESSION HAS ENDED.')
      CALL FINITT(525,125)
C
C     ENTER THE MOVIE OPTION MODE FROM THE SEARCH MODE.
  340 IOP = 1
      CALL ERASE
      WRITE (6,350)
  350 FORMAT (1X,'THE MOVIE MODE HAS BEEN ENTERED')
      GO TO 275
      END


      SUBROUTINE BCKRND(FILMNM,TIME,NFRAME)
C
C     THE BCKRND SUBROUTINE DRAWS THE BACKGROUND OF THE PICTURE
C     CONSISTING OF THE FOLLOWING:
C                (1)  THE FREEWAY
C                (2)  THE RAMP
C                (3)  THE SENSOR AND MERGE POINT LOCATIONS
C                (4)  ALL ALPHANUMERIC OUTPUT INCLUDING THE FRAME NUMBER,
C                     TIME, FILM CODE, AND LABELING.
      DIMENSION IXRL(20),IYRL(20),IXRR(16),IYRR(16)
```

```
      INTEGER FILMNM(2)
      DATA IXRL/683,645,607,570,535,504,476,451,431,419,412,413,446,413,
     *368,398,434,454,466,473/
      DATA IYRL/660,648,632,614,592,566,538,507,472,434,395,355,189,231,
     *285,219,150,91,45,0/
      DATA IXRR/492,437,432,433,441,456,476,504,536,570,608,640,680,758,
     *844,1023/
      DATA IYRR/68,321,359,399,439,476,509,534,563,586,605,622,635,652,
     *661,662/
      CALL MOVABS (0,742)
      CALL DRWABS (1023,742)
      CALL MOVABS(757,672)
      CALL DRWABS(757,678)
      CALL DRWABS(0,678)
      CALL MOVABS(757,672)
      DO 10 I=1,15
   10 CALL DRWABS(IXRL(I),IYRL(I))
      CALL MOVABS(354,274)
      DO 20 I=16,20
   20 CALL DRWABS(IXRL(I),IYRL(I))
      CALL MOVABS(500,0)
      DO 30 I=1,16
   30 CALL DRWABS(IXRR(I),IYRR(I))
      CALL MOVABS(543,742)
      CALL DSHABS(543,678,12)
      CALL MOVABS(343,742)
      CALL DSHABS(343,678,12)
      CALL MOVABS(143,742)
      CALL DSHABS(143,678,12)
      CALL MOVABS(960,742)
      CALL DSHABS(960,662,12)
      CALL HOME
      CALL ANMODE
      WRITE(6,40) (FILMNM(I),I=1,2),NFRAME,TIME
   40 FORMAT (/10X,'F3',12X,'F2',13X,'F1',28X,'MP',/////1X,2A4,5X,1I8,5X
     *,1F9.3)
      RETURN
      END
```

```
      SUBROUTINE GRBDRW(IFIELD)
C
C     THE GRBDRW SUBROUTINE DRAWS THE GREEN BAND DISPLAY.  IF THE LIGHT
C     IS ON A LINE IS DRAWN IF NOT THE BEAM IS MOVED WITHOUT DRAWING.
      DIMENSION IXGB(153),IYGB(153),IFIELD(152)
      DATA IXGB/425,424,423,422,421,420,419,419,418,417,416,415,
     1415,414,414,413,413,412,412,411,411,410,409,409,408,407,407,
     2406,406,405,405,404,403,403,403,403,403,404,404,404,405,
     3405,405,406,406,407,408,408,409,410,411,412,413,414,415,417,418,
     4419,421,422,423,425,427,429,431,433,435,437,439,441,443,445,447,
     5449,452,454,457,460,462,464,467,470,473,476,479,482,485,488,491,
     6494,497,500,503,506,509,512,515,518,521,524,528,532,535,538,541,
     7545,548,552,555,558,562,565,569,573,577,581,585,588,592,596,600,
     8603,606,610,614,618,621,624,628,632,636,640,643,647,651,655,659,
     9663,667,671,675,679,683,687,691,695,699,703,707,711,715,719,724/
      DATA IYGB/238,242,246,250,254,258,262,266,270,273,277,281,285,289,
     1293,297,301,305,309,313,317,321,325,329,333,337,341,345,349,353,
     2357,361,365,369,373,377,381,384,388,392,396,400,404,408,412,416,
     3420,424,428,432,436,440,443,447,451,455,458,462,466,470,474,478,
     4481,484,488,492,496,499,503,506,510,513,516,519,522,526,529,532,
     5535,538,542,545,548,551,554,557,560,562,565,568,571,574,576,579,
     6582,584,586,589,592,594,596,599,602,604,607,609,611,614,616,618,
     7620,622,624,626,629,631,633,634,636,638,639,641,642,644,646,648,
     8649,650,652,654,655,656,657,658,659,660,661,662,663,664,665,666,
     9667,667,668,668,669,669,670,671,671,672,672/
      CALL MOVABS(IXGB(1),IYGB(1))
      DO 3 LITN=1,152
      NBAND=LITN+1
      ITEST=IFIELD(LITN)
      GO TO (1,2), ITEST
    1 CALL MOVABS(IXGB(NBAND),IYGB(NBAND))
      GO TO 3
    2 CALL DRWABS(IXGB(NBAND),IYGB(NBAND))
    3 CONTINUE
      CALL MOVABS(420,236)
      CALL DRWABS(430,240)
      CALL MOVABS(724,668)
```

```
      CALL DRWABS(724,678)
      RFTURN
      END


      SUBROUTINE TRKDRW(IXLOC)
C
C     THIS ROUTINE DRAWS TRUCKS AT IXLOC.
      COMMON/NSTRT/ LSTART
      IY = 686
      IF (LSTART.EQ.11) IY = 714
      IF (IXLOC.LT.50) GO TO 10
      IF (IXLOC.GT.973) GO TO 10
      CALL MOVABS (IXLOC,IY)
      CALL DRWREL(-50,0)
      CALL DRWREL(0,18)
      CALL DRWREL(32,0)
      CALL DRWREL(0,-18)
      CALL MOVREL(16,0)
      CALL WHEEL
      CALL MOVREL(-28,0)
      CALL WHEEL
      CALL MOVREL(-2,0)
      CALL WHEEL
      CALL MOVREL(26,14)
      CALL DRWREL(8,0)
      CALL DRWREL(4,-8)
      CALL DRWREL(6,0)
      CALL DRWREL(0,-6)
   10 RFTURN
      END


      SUBROUTINE CARDRW(IXLOC)
C
C     THIS ROUTINE DRAWS CARS AT IXLOC.
      COMMON/NSTRT/ LSTART
      IY = 686
```

```
      IF (LSTART.EQ.11) IY = 714
      IF (IXLOC.LT.20) GO TO 10
      IF (IXLOC.GT.1003) GO TO 10
      CALL MOVABS (IXLOC,IY)
      CALL DRWREL(-20,0)
      CALL DRWREL(0,3)
      CALL DRWREL(3,3)
      CALL DRWREL(4,1)
      CALL DRWREL(4,0)
      CALL DRWREL(2,-3)
      CALL DRWREL(7,-1)
      CALL DRWREL(0,-3)
      CALL MOVREL(-2,0)
      CALL WHEEL
      CALL MOVREL(-8,0)
      CALL WHEEL
   10 RETURN
      END
```

```
      SUBROUTINE CYLDRW(IXLOC)
C
C     THIS ROUTINE DRAWS MOTORCYCLES AT IXLOC.
      COMMON/NSTRT/ LSTART
      IY = 686
      IF (LSTART.EQ.11) IY = 714
      IF (IXLOC.LT.13) GO TO 10
      IF (IXLOC.GT.1010) GO TO 10
      CALL MOVABS (IXLOC,IY)
      CALL WHEEL
      CALL DRWREL(-5,0)
      CALL WHEEL
      CALL DRWREL(0,1)
      CALL DRWREL(1,1)
      CALL DRWREL(2,0)
      CALL DRWREL(1,1)
      CALL DRWREL(5,0)
      CALL DRWREL(1,-1)
      CALL DRWREL(2,0)



      CALL DRWREL(1,-1)
      CALL DRWREL(0,-1)
   10 RETURN
      END
```

```
      SUBROUTINE WHEEL
C
C     THIS SUBROUTINE DRAWS WHEELS FOR THE VEHICLES.
      CALL DRWREL(0,-1)
      CALL DRWREL(-1,-1)
      CALL DRWREL(-2,0)
      CALL DRWREL(-1,1)
      CALL DRWREL(0,1)
      RETURN
      END
```

```
              ***************************************
              *                                     *
              *                                     *
              *  LISTING OF THE GREEN BAND CONTROL  *
              *        PROGRAM SIMULATION           *
              *                                     *
              *                                     *
              ***************************************
```

```
C     SIMULATED GREEN BAND EXCUTIVE ROUTINE
C
C     ISR  CALLED 500 TIMES/SEC
C     HVP  CALLED 100 TIMES/SEC, 1ST CALL ON SECOND INTERRUPT
C     GRS  CALLED 100 TIMES/SEC, 1ST CALL ON THIRD INTERRUPT
C     GBUD CALLED 100 TIMES/SEC, 1ST CALL ON FOURTH INTERRUPT
C     XVEH AND FRMOUT CALLED TO CORRESPOND TO THE FILM SPEED
```

```
C
      COMMON/GBT/NBANDS,TBAND(15),VBAND(15),XLE(15),XTE(15),
     *          BT1(15),BT2(15),ALLG,VMRG
      COMMON/OCNTRL/  IOUT
      COMMON/RUN/ TSTART,TSTOP
      COMMON/SFLAG/  ACTF(7)
      COMMON/TIMER/ICLOCK,TIME,TNEXT
C
C     FOLLOWING VARIABLE CONTROLS DETAILED OUTPUT OPTION...
      IOUT = 1
      IOUT = 0
C
C     FOLLOWING VARIABLE CONTROLS THE RATE OF FRAME DATA OUTPUT
      NOUT=28
      NOUT=21
C
C     ESTABLISH RUN TIME CONDITIONS...
      TSTART = 0.850
      TSTOP = 2.0
C
C     INITIALIZE TIME VARIABLES
      TIME = TSTART
      ICLOCK = TIME*500.
C
C     INITIALZE VARIABLES FOR ISR AND GBUD ROUTINES...
      NBANDS = 0
      ALLG = 0.
      VMRG = 66.
C
      CALL ISRINI
      CALL HVPINI
C
C     SET UP CALLING SEQUENCE COUNTERS FOR SUBROUTINES
      NHVP = 3
      NGBS = 2
      NGBUD = 1
      NDISPL=0
C
C     MAIN PROCESSING SEQUENCE...
  100 CALL ISR
```

```
C
      NHVP = NHVP +1
      IF(NHVP.EQ.5)  CALL HVP
      IF(NHVP.EQ.5)  NHVP = 0
C
      NGBS = NGBS + 1
      IF(NGBS.EQ.5)  CALL GBS
      IF(NGBS.EQ.5)  NGBS = 0
C
      NGBUD = NGBUD + 1
      IF(NGBUD.EQ.5)  CALL GBUD
      IF(NGBUD.EQ.5)  NGBUD = 0
C
      NDISPL=NDISPL+1
      IF  (NDISPL.EQ.NOUT) CALL XVEH
      IF  (NDISPL.EQ.NOUT) CALL FRMOUT
      IF  (NDISPL.EQ.NOUT) NDISPL=0
C
C     UPDATE CLOCK (COUNT OF 2 MILLISECOND INTERRUPT CYCLES)
      ICLOCK = ICLOCK + 1
      CLOCK = ICLOCK
      TIME = CLOCK/500.0
      IF(TIME.LT.TSTOP)  GO TO 100
      STOP
      END


      SUBROUTINE ISRINI
      COMMON/INPUT/  RAWDAT(7,100,5),STNEXT(7),INEXT(7)
      COMMON/NEXT/ISENS,T1,ICAR,VEL,XLEN,ETA
      COMMON/RUN/ TSTART,TSTOP
      COMMON/SFLAG/  ACTF(7)
      COMMON/TIMER/ICLOCK,TIME,TNEXT
      DIMENSION INSENS(7)
      DIMENSION ADJUST(7)
      DATA ADJUST/0.0,1.839,1.299,-1.55,-4.231,2.93,1.536/
C
C     CLEAR THE RAWDAT ARRAY
      DO 100 I = 1,7
```

```
      DO 100 J = 1,100
      DO 100 K = 1,5
  100 RAWDAT(I,J,K) = 0.0
C
C     CLEAR VEHICLE COUNTERS FOR INPUT ARRAY...
      DO 1 I = 1,7
    1 INSENS(I) = 0
C
C     BRING IN ENOUGH FREEWAY DATA FROM CARDS TO SATISFY RUN TIMING
C     PARAMETERS ESTABLISHED IN MAIN...
C
    2 READ(5,200) ISENS,T2,ICAR,VEL,XLEN,ETA
      IF (T2.LT.TSTART) GO TO 2
  200 FORMAT(9X,I1,F12.0,I12,3F12.0)
C     TIME READ IS T2, ADJUST TO T1 VALUE...
      T1 = T2 - (XLEN+6.)/VEL
C     CONVERT VEHICLE LENGTH IN FEET TO LENGTH IN TIME...
      XLEN = XLEN/VEL
C
C     ADJUST THE VEHICLE VELOCITY.
      VEL = VEL + ADJUST(ISENS)
      WRITE(6,101) TIME,ISENS,T1,ICAR,VEL,XLEN,ETA
  101 FORMAT(  1X,'TIME=',F9.3,',ISRINI READING...',I3,F9.3,I4,3F9.3)
C
C
C     COUNT NUMBER OF VEHICLES READ AT EACH SENSOR STATION...
      INSENS(ISENS) = INSENS(ISENS) + 1
      IF(INSENS(ISENS).GT.100)  STOP
C
C     LOAD RAW DATA ARRAY FROM PUNCHED TOGAP OUTPUT...
         RAWDAT(8-ISENS,INSENS(ISENS),1) = ETA
         RAWDAT(8-ISENS,INSENS(ISENS),2) = T1
         RAWDAT(8-ISENS,INSENS(ISENS),3) = VEL
         RAWDAT(8-ISENS,INSENS(ISENS),4) = XLEN
         RAWDAT(8-ISENS,INSENS(ISENS),5) = ICAR
C
      IF(T1.LT.(TSTOP+15.))  GO TO 2
C
C     DATA IS READY, SET UP RUN CONTROL FLAGS AND TIMERS...
      DO 10 I = 1,7
```

```
      STNEXT(I) = RAWDAT(I,1,2)
   10 INEXT(I) = 1
C
C     TURN OFF 'SENSOR DATA READY' FLAGS FOR HVP...
      DO 50 I = 1,7
   50 ACTF(I) = +1.0
C
      RETURN
      END


      SUBROUTINE ISR
      COMMON/INPUT/  RAWDAT(7,100,5),STNEXT(7),INEXT(7)
      COMMON/NEXT/ISENS,T1,ICAR,VEL,XLEN,ETA
      COMMON/SFLAG/  ACTF(7)
      COMMON/TIMER/ICLOCK,TIME,TNEXT
C
C     THIS SUBROUTINE SIMULATES THE SCANNING OF FIELD SENSOR DATA
C     IN THE ON-LINE SYSTEM BY USING HISTORICAL DATA OBTAINED FROM
C     REDUCED DATA GENERATED BY THE TOGAP SYSTEM.
C
C     NOTE...THE PROGRAM DOES NOT LOOK AHEAD IN TIME.
C     THE VARIABLE TNEXT INDICATES WHEN THE NEXT FIELD SENSOR
C     DATA WILL BE READY.  UNTIL THAT TIME, ISR REPORTS NO NEW DATA.
C
C     SCAN SENSORS, TURN ON 'DATA READY' FLAGS IF APPROPRIATE...
      DO 1000 I = 1,7
      IF(TIME.GE.STNEXT(I))  ACTF(I) = -1.0
 1000 CONTINUE
C
C     SEARCH FOR SMALLEST OF THE NEXT SENSOR ACTIVATION TIMES...
C     NOTE... THIS VALUE IS NOT CURRENTLY USED IN THE SIMULATION,
C     BUT COULD BE USED IN A SIMULATION STRATEGY BASED ON EVENT
C     SCANNING RATHER THAN THE CURRENT MODE OF TIME SCANNING.
C
      TNEXT = 1.0E60
      DO 2000 I = 1,7
      IF(STNEXT(I).LT.TNEXT) TNEXT=STNEXT(I)
 2000 CONTINUE
```

```
C
      RETURN
      END


      SUBROUTINE HVPINI
      COMMON/BINS/  SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *              SNC(9,7),NLIST(7)
      COMMON/FWYDAT/NVEH,FETA(63),FT1(63),FVEL(63),FXLEN(63),NC(63)
C
C     THIS ROUTINE CLEARS THE HIGHWAY VEHICLE DATA LIST COUNTS TO
C     ZERO VALUES TO SET UP THE INITIAL HIGHWAY VEHICLE PROCESSING
C     CONDITIONS.
C
      DO 50 I = 1,7
   50 NLIST(I) = 0
C
      NVEH = 0
C
      RETURN
      END


      SUBROUTINE HVP
      COMMON/BINS/  SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *              SNC(9,7),NLIST(7)
      COMMON/FWYDAT/NVEH,FETA(63),FT1(63),FVEL(63),FXLEN(63),NC(63)
      COMMON/NEXT/ISENS,T1,ICAR,VEL,XLEN,ETA
      COMMON/OCNTRL/  IOUT
      COMMON/SFLAG/  ACTF(7)
      COMMON/TIMER/ICLOCK,TIME,TNEXT
      COMMON /EXPT/ GBDATA
C
C     NOTE... THOLD CONTROLS FWY VEHICLE RETENTION TIME IN DATA TABLES...
      DATA THOLD / 20.48 /
C
C     THIS ROUTINE SIMULATES THE ACTIONS OF THE HVP ROUTINE IN
C     THE ON-LINE GREEN BAND CONTROL PROGRAM.   NOTE THAT THE
```

```
C     VARIABLES IT USES TO UPDATE THE FREEWAY LISTS ARE ALREADY
C     COMPUTED AND SUPPLIED BY ISR.  IN THE ON-LINE SYSTEM, THESE
C     VALUES MUST BE COMPUTED BY THE HVP FROM SENSOR EVENT TIME DATA.
C
C     PROCESS CURRENT FWY DATA LISTS TO REMOVE ANY VEHICLES
C     PAST DUE AT THE MERGE POINT...
C
      DO 200 ILIST = 1,7
      JNV = NLIST(ILIST)
      IF(JNV.EQ.0) GO TO 200
      NSQ = 0
      DO 100 JVEH = 1,JNV
C     MARK OVERTIME VEHICLES WITH NEGATIVE ETA VALUE...
      IF(SETA(JVEH,ILIST).GE.TIME)   GO TO 100
         SETA(JVEH,ILIST) = -99.
         NSQ = NSQ + 1
  100 CONTINUE
C     ROUTINE 'SQUEZE' REMOVES ALL VEHICLES IN CURRENT LIST WITH
C     NEGATIVE ETA VALUES...
      IF(NSQ.GT.0)  CALL SQUEZE(ILIST)
  200 CONTINUE
C
C     DO LIST SCAN FOR NEW DATA UPDATES...
      GRDATA = -1.
      GRDATA = 1.
      DO 400 I = 1,7
C
C     IS THERE NEW DATA READY FOR THIS LIST?
      IF(ACTF(I).GE.0.) GO TO 400
C
C     DATA IS AVAILABLE FOR LIST I, IS THE LIST CURRENTLY EMPTY?
  401 IF(NLIST(I).GT.0)  GO TO 405
C
C     THE CURRENT LIST IS EMPTY, SO INSERT THE NEW DATA AND CONTINUE
C     PROCESSING WITH THE NEXT LIST...
      CALL INSERT(I)
      GO TO 400
C
C     THE CURRENT LIST HAS AT LEAST ONE ENTRY.
C
```

```
C
C     CHECK VEHICLE RETENTION TIME FOR THE TOP VEHICLE IN LIST
C     (E.G., IS THE VEHICLE OVERDUE AT THE NEXT DOWNSTREAM SENSOR?)
C
  405 IF((ST1(1,I)+THOLD).GT.TIME)  GO TO 410
C
C     FIRST VEHICLE IS OVERDUE, DELETE IT AND PUSH LIST UP...
      SETA(1,I) = -99.
      CALL SQUEZE(I)
C     PUT NEW DATA IN UPDATED LIST...
      CALL INSERT(I)
C
C     FIRST VEHICLE IN THE LIST IS NOT OVERDUE, CHECK TO SEE IF THE LIST
C     IS FULL...
  410 IF(NLIST(I).EQ.9)  GO TO 420
C
C     LIST IS NOT FULL, STORE THE NEW DATA...
      CALL INSERT(I)
      GO TO 400
C
C     LIST IS NOW FULL, DELETE FIRST VEHICLE...
  420 SETA(1,I) = -99.
      CALL SQUEZE(I)
C     LIST NOW HAS 8 ENTRIES, STORE THE NEW DATA IN THE LAST POSITION...
      CALL INSERT(I)
  400 CONTINUE
C
C     NOW FINISHED WITH INDIVIDUAL SENSOR LIST PROCESSING, SO COPY
C     THE INDIVIDUAL SENSOR BINS INTO THE MASTER ARRAY USED BY THE SIMULATED
C     GRS ROUTINE...
      IGO = 1
      DO 500 ILIST = 1,7
      NV = NLIST(ILIST)
      IF(NV.EQ.0)  GO TO 500
      DO 501 ICOPY = 1,NV
         FETA(IGO) = SETA(ICOPY,ILIST)
         FT1  (IGO) = ST1  (ICOPY,ILIST)
         FVEL(IGO) = SVEL(ICOPY,ILIST)
         FXLEN(IGO)= SLEN(ICOPY,ILIST)
         NC   (IGO) = SNC(ICOPY,ILIST)
         IGO = IGO + 1
```

```
  501 CONTINUE
  500 CONTINUE
C
C     COMPUTE NUMBER OF VEHICLES IN CURRENT LIST...
      NVEH = IGO - 1
      IF(JOUT.EQ.1)  CALL BINOUT(1)
      RETURN
      END


      SUBROUTINE INSERT(ILIST)
      COMMON/BINS/  SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *              SNC(9,7),NLIST(7)
      COMMON/INPUT/  RAWDAT(7,100,5),STNEXT(7),INEXT(7)
      COMMON/NEXT/ISENS,T1,ICAR,VEL,XLEN,ETA
      COMMON/SFLAG/  ACTF(7)
      DATA HMIN/0.8/
C
C     THIS ROUTINE PLACES NEW FREEWAY VEHICLE DATA IN THE CURRENT DATA
C     LIST REPRESENTING THE SENSOR 'BIN' AT LOCATION ILIST IN THE ON-LINE
C     PROGRAM.
C
C     IF CURRENT LIST EMPTY, GO TO SIMPLE INSERTION CASE...
      IF(NLIST(ILIST).EQ.0)  GO TO 200
C
C     CURRENT LIST IS NOT EMPTY, PUT NEW DATA AT END...
  100 SETA(NLIST(ILIST)+1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),1)
      ST1 (NLIST(ILIST)+1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),2)
      SVEL(NLIST(ILIST)+1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),3)
      SLEN(NLIST(ILIST)+1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),4)
      SNC (NLIST(ILIST)+1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),5)
C
C     NOW DO MINIMUM FWY VEHICLE ETA TEST ... E.G., IF COMPUTED MP ETA
C     IS LESS THAN THAT OF THE PREVIOUS VEHICLE, SET THE NEW ETA EQUAL
C     TO THE OLD VALUE + HMIN.
C
      IF(SETA(NLIST(ILIST)+1,ILIST).LT.SETA(NLIST(ILIST),ILIST))
     *     SETA(NLIST(ILIST)+1,ILIST) = SETA(NLIST(ILIST),ILIST) + HMIN
C     BUMP LIST COUNT...
```

```
      NLIST(ILIST) = NLIST(ILIST) + 1
      GO TO 2000
C
C     LIST WAS EMPTY, INSERT NEW DATA AT FIRST POSITION...
  200 SETA(1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),1)
      ST1 (1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),2)
      SVEL(1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),3)
      SLEN(1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),4)
      SNC (1,ILIST) = RAWDAT(ILIST,INEXT(ILIST),5)
C
C     ADJUST LIST COUNT TO REFLECT SINGLE ENTRY...
      NLIST(ILIST) = 1
C
C     TURN OFF 'SENSOR ACTIVE' FLAG AND UPDATE POINTER VARIABLES TO
C     FUTURE SENSOR ACTIVATIONS ON THE CHANNEL...
C
 2000 ACTF(ILIST) = +1.0
      INEXT(ILIST) = INEXT(ILIST) + 1
      STNEXT(ILIST) = RAWDAT(ILIST,INEXT(ILIST),2)
C
      RETURN
      END


      SUBROUTINE SQUEZE(ILIST)
      COMMON/BINS/  SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *              SNC(9,7),NLIST(7)
C
C     THIS ROUTINE COMPRESSES FREEWAY DATA LISTS BY MOVING TABLE
C     ENTRIES UP IN THE SPECIFIED LIST TO REPLACE SENSOR REPORTS
C     MARKED AS UNACCEPTABLE BY THE HVP.
C
C     THE 'DATA UNACCEPTABLE' CONDITION FOR THE SIMULATION SCHEME IS
C     INDICATED BY A NEGATIVE VALUE FOR A VEHICLE ETA SET BY HVP.
C
      IF(NLIST(ILIST).GT.1)  GO TO 200
C
C     LIST HAD ONLY ONE ELEMENT, DELETE BY ZEROING COUNT.
      NLIST(ILIST) = 0
```

```
      RETURN
C
C     FIND FIRST NEGATIVE ETA IN CURRENT LIST...
  200 NNLIST = NLIST(ILIST)
      DO 205 IBAD = 1,NNLIST
      IF(SETA(IBAD,ILIST).LT.0.)  GO TO 207
  205 CONTINUE
C
C     LIST IS NOW O.K., RETURN TO HVP...
      RETURN
C
C     SQUEEZE LIST UP FROM BAD DATA ROW...
  207 IF(IBAD.EQ.NNLIST)  GO TO 209
      INM1 = NNLIST - 1
      DO 208 IMOVE = IBAD,INM1
         SETA(IMOVE,ILIST) = SETA(IMOVE+1,ILIST)
         ST1 (IMOVE,ILIST) = ST1 (IMOVE+1,ILIST)
         SVEL(IMOVE,ILIST) = SVEL(IMOVE+1,ILIST)
         SLEN(IMOVE,ILIST) = SLEN(IMOVE+1,ILIST)
         SNC (IMOVE,ILIST) = SNC (IMOVE+1,ILIST)
  208 CONTINUE
      NLIST(ILIST) = NLIST(ILIST) - 1
      IF(NLIST(ILIST).EQ.0)  RETURN
      GO TO 200
  209 NLIST(ILIST) = NLIST(ILIST) - 1
      RETURN
      END


      SUBROUTINE BINOUT(IOPT)
      COMMON/BINS/  SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *              SNC(9,7),NLIST(7)
      COMMON/FWYDAT/NVEH,FETA(63),FT1(63),FVEL(63),FXLEN(63),NC(63)
      COMMON/TIMER/ICLOCK,TIME,TNEXT
C
C     THIS ROUTINE PRINTS THE CURRENT CONTENTS OF THE SEVEN HIGHWAY
C     VEHICLE DATA BINS IN THE SIMULATED HVP ROUTINE.
C
      IF(IOPT.EQ.1)  WRITE(6,10)  TIME,NVEH
```

```
   10 FORMAT(/  ,' HVP FINISHED AT TIME = ',F9.3,', WITH NVEH = ',I4)
C
      DO 100 IS = 1,7
      WRITE(6,90)   IS,IS,NLIST(IS)
   90 FORMAT(1X,'SENSOR BIN ',I2,'... NLIST(',I2,')=',I2)
      IF(NLIST(IS).EQ.0)  GO TO 100
      NV = NLIST(IS)
      DO 95 JV = 1,NV
   95 WRITE(6,96)   JV,SETA(JV,IS),ST1(JV,IS),SVEL(JV,IS),
     *              SLEN(JV,IS),SNC(JV,IS)
   96 FORMAT(1X,'VEH. NO. = ',I2,', ETA,T1,VEL,LEN,CARID=',4F10.3,F5.0)
  100 CONTINUE
      RETURN
      END


      SUBROUTINE GBS
      COMMON/BINS/   SETA(9,7),ST1(9,7),SVEL(9,7),SLEN(9,7),
     *               SNC(9,7),NLIST(7)
      COMMON/FWYDAT/NVEH,FETA(63),FT1(63),FVEL(63),FXLEN(63),NC(63)
      COMMON/GBT/NBANDS,TBAND(15),VBAND(15),XLE(15),XTE(15),
     *          BT1(15),BT2(15),ALLG,VMRG
      COMMON/OCNTRL/   IOUT
      COMMON/TIMER/ICLOCK,TIME,TNEXT
      COMMON /EXPT/ GBDATA
      REAL*4  THDY,LHDY
      DATA THDY,LHDY /-1.2,0.4/
      DATA ZERO /0.0/
C
C     THIS ROUTINE SIMULATES THE GBS ROUTINE OF THE ON-LINE CONTROL
C     SYSTEM.  FREEWAY VEHICLE DATA IS SEARCHED FOR ACCEPTABLE
C     GAPS.  BANDS REPRESENTING THESE GAPS ARE CREATED FOR OUTPUT BY
C     THE GRUD ROUTINE.
C
C     PRINT FREEWAY VEHICLE DATA TABLES...
      IF(IOUT.EQ.0)   GO TO 8891
      WRITE(6,8888)   TIME,NVEH,NLIST
 8888 FORMAT(  1X,'TIME=',F9.3,', NVEH = ',I3,', NLIST= ',7I5)
      IF(NVEH.EQ.0)   GO TO 8891
```

```
      DO 8889 IV = 1,NVEH
 8889 WRITE(6,8890)  IV,FETA(IV),FT1(IV),FVEL(IV),FXLEN(IV),NC(IV)
 8890 FORMAT(1X,'FWY VEH NO. ',I2,' ...',4F12.3,I5)
 8891 CONTINUE
C     ABORT THE RUN IF DATA TABLES ARE OVERFLOWING...
      IF(NVEH.GT.63)  STOP
C
C     TEST FOR UPSTREAM SECTION 'ALL CLEAR' CONDITION...
      IF(NVEH.GT.0)  GO TO 100
      ALLG = 1.
      RETURN
C
C     SET BAND COUNTER TO ZERO, CREATE NEW SET...
  100 CONTINUE
      IF (GBDATA.LT.0.) RETURN
      NBANDS = 0
C     TURN OFF ALL GREEN FLAG TO INDICATE THERE ARE ACTIVE BANDS...
      ALLG = 0.
C
C     CREATE FIRST BAND
      NVSUM = 0
      DO 10 NSENS = 1,7
      NVSUM = NVSUM + NLIST(NSENS)
      IF (NLIST(NSENS).EQ.0) GO TO 10
      GO TO 11
   10 CONTINUE
   11 NL = 1
      NT = 1
      NBANDS = NBANDS + 1
      XLE(NBANDS) = 844.
      CALL XLTE(FETA(1),ZERO,THDY,VBAND(NBANDS),XTE(NBANDS))
      TBAND(NBANDS) = TIME
      BT1(NBANDS) = -1.
      BT2(NBANDS) = FETA(1)
      IF(NVEH.EQ.1)  GO TO 300
C
C     DO INTERMEDIATE BAND PROCESSING...
  200 NT = NT + 1
      IF(NT.GT.NVEH)  GO TO 300
      IF (NT.LE.NVSUM) GO TO 210
```

```
  220 NSENS = NSENS + 1
      NVSUM = NVSUM + NLIST(NSENS)
      IF (FETA(NT).GT.(FETA(NL)+0.25)) GO TO 210
      GO TO 200
C     CHECK FOR MIN ETA SPACING OF 2 SECONDS
  210 IF (FETA(NT).GT.(FETA(NL)+2.0)) GO TO 250
      NL = NT
      GO TO 200
C     COMPUTE MEASURED GAP
  250 XMGAP = FETA(NT) - FETA(NL) - FXLEN(NL)
C     COMPUTE REQUIRED GAP
      CALL RAYGAP(NT,NL,RGAP)
      IF(IOUT.EQ.1)  WRITE(6,2000)  TIME,NT,NL,XMGAP,RGAP
 2000 FORMAT(  1X,'TIME=',F9.3,', TVEH,LVEH,MGAP,RGAP...',2I3,2F12.3)
C     IS THIS GAP ADEQUATE?
      IF(XMGAP.GE.RGAP)  GO TO 275
      NL = NT
      GO TO 200
C
C     YES, CREATE NEW BAND...
  275 NBANDS = NBANDS + 1
      CALL XLTE(FETA(NL),FXLEN(NL),LHDY,VBAND(NBANDS),XLE(NBANDS))
      CALL XLTE(FETA(NT),ZERO,THDY,VDUMMY,XTE(NBANDS))
      TBAND(NBANDS) = TIME
      BT1(NBANDS) = FETA(NL)
      BT2(NBANDS) = FETA(NT)
      NL = NT
      GO TO 200
C
C     DO PROCESSING FOR LAST BAND
  300 NBANDS = NBANDS +1
      CALL XLTE(FETA(NVEH),FXLEN(NVEH),LHDY,VBAND(NBANDS),XLE(NBANDS))
      XTE(NBANDS) = -9999.
      TBAND(NBANDS) = TIME
      BT1(NBANDS) = FETA(NVEH)
      BT2(NBANDS) = -1.0
C
 1000 CONTINUE
      IF(IOUT.EQ.0)  RETURN
      WRITE(6,1001) TIME,NBANDS
```

```
 1001 FORMAT(  1X,'TIME=',F9.3, ',GBS FINISHED WITH NBANDS = ',I5)
      IF(NBANDS.EQ.0)  RETURN
      DO 1002 J = 1,NBANDS
 1002 WRITE(6,1003)J,TBAND(J),VBAND(J),XTE(J),XLE(J),BT1(J),BT2(J)
 1003 FORMAT(1X,'BAND NO. ',I2,' ...',6F12.3)
      RETURN
      END


      SUBROUTINE XLTE(ZETA,ZLEN,ZLTHDY,VNOW,EDGE)
      COMMON/GBT/NBANDS,TBAND(15),VBAND(15),XLE(15),XTE(15),
     *            BT1(15),BT2(15),ALLG,VMRG
      COMMON/TIMER/ICLOCK,TIME,TNEXT
      DATA ACC/3.0/
C
C     ROUTINE TO SIMULATE BAND POSITIONING LOGIC FOR TAMPA VARIABLE
C     GREEN BAND SPEED CASE.
C
      UFTMP = ZETA + ZLTHDY + ZLEN - TIME
      WORK1 = (VMRG*VMRG - 1936.)/6.
      PAM = (580. - WORK1)/VMRG
      UFTAE = UFTMP - PAM
      UFTSL = UFTAE - ((VMRG-44.)/ACC)
      UFTDB = UFTSL - 6.
C
      IF(UFTMP.LT.0.)  GO TO 100
      IF(UFTAE.LT.0.)  GO TO 200
      IF(UFTSL.LT.0.)  GO TO 300
C
C     BAND EDGE IDENTIFIED AS BEING IN 30 MPH SECTION...
C     SIMULATION OF ON-LINE CODE AT LABEL SLOW...
      VNOW = 44.
      EDGE = -UFTDB * 44.
      RETURN
C
C     SIMULATION OF ON-LINE CODE AT LABEL ATMP...
  100 VNOW = VMRG
      EDGE = 844.
      RETURN
```

```
C
C     SIMULATION OF ON-LINE CODE AT LABEL FAST...
  200 VNOW = VMRG
      EDGE = 844. - (UFTMP*VMRG)
      RETURN
C
C     SIMULATION OF ON-LINE CODE AT LABEL UFACC...
  300 VNOW = 44. - (UFTSL*3.)
C     SET BAND 'STILL ACCELERATING' CONDITION...
      VNOW = - VNOW
      EDGE = 264. + (VNOW*VNOW - 1936.)/6.
      RETURN
      END


      SUBROUTINE RAYGAP(ITRAIL,ILEAD,RGAP)
      COMMON/FWYDAT/NVEH,FETA(63),FT1(63),FVEL(63),FXLEN(63),NC(63)
      COMMON/GBT/NBANDS,TBAND(15),VBAND(15),XLE(15),XTE(15),
     *           BT1(15),BT2(15),ALLG,VMRG
      DATA HZERO /0.8/
      DATA DEE /4.0/
      DATA TLRAMP/0.3/
C
C     RAYGAP IMPLEMENTS THE ALGORITHM USED BY RAYTHEON TO COMPUTE
C     THE REQUIRED GAP SIZE FOR GREEN BAND GENERATION AS A FUNCTION
C     OF THE VELOCITIES OF THE LEADING AND TRAILING FREEWAY VEHICLES
C     DEFINING A GAP, AND THE VELOCITY OF THE RAMP VEHICLE BEING
C     CONSIDERED FOR THE GAP.  SINCE THE GREEN BAND RAMP VEHICLES ARE
C     NOT TRACKED BY THE ON-LINE SYSTEM, THE CURRENT GREEN BAND SPEED
C     IS USED FOR THE RAMP VEHICLE SPEED.
C
C     RGAP = (H1+H1P+HZERO) + (H2+H2P+HZERO) + TLRAMP
C
      VT = FVEL(ITRAIL)
      VL = FVEL(ILEAD)
C
      IF(VT.GT.VL.AND.VMRG.GT.VL)  GO TO 100
      IF(VT.GT.VL.AND.VL.GT.VMRG)  GO TO 200
      IF(VL.GT.VT.AND.VT.GT.VMRG)  GO TO 300
```

```
      IF(VL.GT.VT.AND.VMRG.GT.VT.AND.VMRG.GT.VL)   GO TO 400
      IF(VL.GT.VT.AND.VMRG.GT.VT.AND.VL.GT.VMRG)   GO TO 500
      RGAP = 1.0E50
      RETURN
C
  100 H1 = (VMRG-VL)*(VMRG-VL)/(2.0*DEE*VL)
      H2 = (VT-VL)*(VT-VL)/(2.0*DEE*VL)
      GO TO 600
C
  200 H1 = 0.
      H2 = ((VT-VL)*(VT-VL)/(2.0*DEE*VL)) + F2(VMRG,VL)
      GO TO 600
C
  300 H1 = 0.
      H2 = F2(VMRG,VT)
      GO TO 600
C
  400 H1 = (VMRG-VL)*(VMRG-VL)/(2.0*DEE*VL)
      H2 = 0.
      GO TO 600
C
  500 H1 = 0.
      H2 = 0.
      GO TO 600
C
C     COMPUTE ETA PREDICTION ERROR ALLOWANCE...
  600 H1P = 0.777*F3(VL)*(FETA(ILEAD)-FT1(ILEAD))
      H2P = 0.777*F3(VT)*(FETA(ITRAIL)-FT1(ITRAIL))
C
      RGAP = H1 + H1P + H2 + H2P + 2.0*HZERO + TLRAMP
      RETURN
      END


      FUNCTION F2(V1,V2)
C
C     THIS FUNCTION COMPUTES THE TIME TO ACCELERATE FROM V1 TO V2
C     IN SECONDS.  V1 AND V2 ARE TO BE SUPPLIED IN FT/SEC.
C
```

```
      VZERO = 146.7
      ALPHA = 5.
      VDIFF = V2 - V1
      TERM = (VZERO-V2)/(VZERO-V1)
      FTIME = (VZERO/(ALPHA*V2))
      FTIME = FTIME*(VDIFF+(VZERO-V2)*ALOG(TERM))
C
      ATIME = 0.11*VDIFF - 0.925
      IF(VDIFF.LE.17.5)  ATIME = 0.057*VDIFF
      IF(VDIFF.GE.41.)  ATIME = 0.16*VDIFF - 2.85
C
C     IF THE 'TRUE' FORMULA FOR F2 IS TO BE USED, SET F2 EQUAL TO
C     THE VARIABLE FTIME IN TTHE FOLLOWING ASSIGNMENT.
C
      F2 = ATIME
      RETURN
      END


      FUNCTION F3(VEL)
C
C     F3(VEL) COMPUTES A TERM IN THE EQUATION USED BY RAYTHEON
C     TO PREDICT FWY VEHICLE ETA UNCERTAINTY IN THE REQUIRED GAP
C     CALCULATIONS.
C
C     VEL IS SUPPLIED IN UNITS OF FT/SEC
C
      F3 = 0.01 + ((88.-VEL)/73.5)
C
      IF(VEL.LT.14.7)  F3 = 0.10
C
      IF(VEL.GT.88.0)  F3 = 0.01
C
      RETURN
      END


      SUBROUTINE GBUD
```

```
      COMMON/GBT/NBANDS,TBAND(15),VBAND(15),XLE(15),XTE(15),
     *          BT1(15),BT2(15),ALLG,VMRG
      COMMON/TIMER/ICLOCK,TIME,TNEXT
      COMMON/LIGHTS/FIELD(160)
      INTEGER*2 FIELD
      REAL*4 ACCEL /3.0/
      DATA DT /0.010/
C
C     THIS ROUTINE SIMULATES THE GBUD ROUTINE IN THE ON-LINE SYSTEM.
C     DATA IN THE CURRENT GREEN BAND TABLE IS UPDATED AND DISPLAYED
C     ON THE SIMULATED FIELD LIGHTS.
C
      IF(ALLG.LT.1)  GO TO 100
      DO 50 I=1,160
   50 FIELD(I)=2
      GO TO 1000
C
C     CLEAR BAND DISPLAY FOR NEW DATA...
  100 DO 101 I=1,160
  101 FIELD(I)=1
C
C     SCAN BAND TABLE
      IF(NBANDS.EQ.0)  GO TO 1000
      DO 105 K = 1,NBANDS
C     DO NOT PROCESS BAND IF XTE IS FARTHER ALONG DISPLAY THAN XLE...
      IF(XTE(K).GE.XLE(K))  GO TO 105
C
C     CONSTANT VELOCITY BAND?
      IF(VBAND(K).GT.0.)  GO TO 150
C
C     NO, STILL ACCELERATING...
      VB = ABS(VBAND(K)) + ACCEL*DT
      IF(VB.GE.VMRG)  VBAND(K) = VMRG
      IF(VB.LT.VMRG)  VBAND(K) = -VB
C
C     UPDATE LEADING AND TRAILING EDGES OF BAND...
  150 DX = VB * DT
      XTE(K) = XTE(K) + DX
      XLE(K) = XLE(K) + DX
C
```

```
      IGO=XTE(K)/4.0
      ISTOP=XLE(K)/4.0
C
      IF(IGO.GT.160) GO TO 105
      IF(ISTOP.LE.0) GO TO 105
      IF(ISTOP.GT.160) ISTOP=160
      IF(IGO.LE.0) IGO=1
      IF(IGO.GE.ISTOP)  GO TO 105
C
      DO 160 I = IGO,ISTOP
 160  FIELD(I)=2
  105 CONTINUE
C
 1000 RETURN
      END


      SUBROUTINE XVEH
      COMMON/INPUT/ RAWDAT(7,100,5),STNEXT(7),INEXT(7)
      COMMON/XLOCAL/ NTKLOC(20),NCRLOC(20),NCYLOC(20)
      COMMON/TIMER/ ICLOCK,TIME,TNEXT
      INTEGER*2  NTKLOC,NCRLOC,NCYLOC,IFLAG(780)
      DIMENSION XSNS(4)
      DATA  XSNS / 543.0,343.0,143.0,-57.0 /
C
C     THIS ROUTINE PLOTS THE VEHICLE LOCATIONS USING TWO METHODS.
C     METHOD 1 MAINTAINS THE VELOCITIES MEASURED AT THE SENSORS UNTIL
C     THE NEXT SENSOR ENTRY OCCURS.  METHOD 2 USES LINEAR INTERPOLATION
C     TO ACHIEVE SMOOTH VEHICLE MOTION BETWEEN THE SENSORS.
C
C     SET VEHICLE FLAGS TO ZERO.
      DO 10 N = 1,780
   10 IFLAG(N) = 0
C
C     INITIALIZE VEHICLE COUNTERS AND LOCATIONS.
      NTRK = 0
      NCAR = 0
      NCYL = 0
      DO 20 N = 1,20
```

```
      NTKLOC(N) = 9999
      NCRLOC(N) = 9999
   20 NCYLOC(N) = 9999
C
C     START AT SENSOR F1 ENTRIES.
      I = 1
C
C     RETURN WHEN LISTS FOR SENSORS F1 THROUGH F4 HAVE BEEN EXAMINED.
   30 IF (I.GT.4) RETURN
C
C     SET LIST ENTRY COUNTER.
      LCNT = 1
C
C     HAS THIS ENTRY OCCURRED?
   40 IF (TIME.GE.RAWDAT(I,LCNT,2)) GO TO 50
C     NO. GO TO THE NEXT SENSOR LIST.
      I = I + 1
      GO TO 30
C     YES. WHAT IS THE VEHICLE NUMBER?
   50 IVEH = RAWDAT(I,LCNT,5)
C
C     HAS THE VEHICLE BEEN PROCESSED?
      IF (IFLAG(IVEH).EQ.1) GO TO 80
C     NO. CALCULATE THE VEHICLE LENGTH.
      XLENTH = (RAWDAT(I,LCNT,4)) * (RAWDAT(I,LCNT,3))
C
C     CLASSIFY THE VEHICLE ACCORDING TO LENGTH.
      ITYPE = 3
      IF (XLENTH.GT.10.0) ITYPE = 2
      IF (XLENTH.GT.24.0) ITYPE = 1
      IF (I.EQ.1) GO TO 90
C
C     SEARCH FOR THE VEHICLES ENTRY IN THE NEXT SENSOR LIST.
      DO 60 ICNT = 1,100
      NCNT = ICNT
      NVEH = RAWDAT(I-1,ICNT,5)
C     IS THIS THE PROPER ENTRY?
      IF (IVEH.EQ.NVEH) GO TO 100
   60 CONTINUE
C
```

```
C
C     NO ENTRY AT THE NEXT SENSOR. PLOT UNTIL THE MIDPOINT OF THE REGION
C     USING METHOD 1.
      MIDPT = XSNS(I) + 100.0
      IXLOC = (RAWDAT(I,LCNT,3)) * (TIME - RAWDAT(I,LCNT,2)) + XSNS(I) +
     *0.5
      IF (IXLOC.GT.MIDPT) GO TO 150
      KXLOC = IXLOC
      GO TO 110
C
C     GO TO THE NEXT ENTRY.
   80 LCNT = LCNT + 1
      GO TO 40
C
C     AN F1 ENTRY.  PLOT UNTIL THE VEHICLE IS OUT OF THE DISPLAYED
C     REGION USING METHOD 1.
   90 IXLOC = (RAWDAT(I,LCNT,3)) * (TIME - RAWDAT(I,LCNT,2)) + XSNS(I) +
     *0.5
      IF (IXLOC.GT.1023) GO TO 150
      KXLOC = IXLOC
      GO TO 110
C
C     AN F2 THROUGH F4 ENTRY WITH AN ENTRY AT THE NEXT SENSOR.  PLOT
C     USING BOTH METHODS.
  100 IXLOC = 0.5 + XSNS(I) + 200.0 * (TIME - RAWDAT(I,LCNT,2)) /
     *(RAWDAT(I-1,NCNT,2) - RAWDAT(I,LCNT,2))
      KXLOC = (RAWDAT(I,LCNT,3)) * (TIME - RAWDAT(I,LCNT,2)) + XSNS(I) +
     *0.5
C
C     RECORD THE VEHICLE LOCATION ACCORDING TO THE VEHICLE TYPE.
  110 GO TO (120,130,140), ITYPE
  120 NTRK = NTRK + 1
      NTKLOC(NTRK) = KXLOC
      NTKLOC(NTRK + 10) = IXLOC
      GO TO 150
  130 NCAR = NCAR + 1
      NCRLOC(NCAR) = KXLOC
      NCRLOC(NCAR + 10) = IXLOC
      GO TO 150
  140 NCYL = NCYL + 1
      NCYLOC(NCYL) = KXLOC
```

```
      NCYLOC(NCYL + 10) = IXLOC
C
C     SET THE VEHICLE FLAG.
  150 IFLAG(IVEH) = 1
      GO TO 80
      END


      SUBROUTINE FRMOUT
C     THIS SUBROUTINE WRITES THE DATA REQUIRED TO DRAW A PICTURE INTO A
C     PREVIOUSLY ALLOCATED DATA FILE.  THE DATA INCLUDES:
C             (1)  THE GREEN BAND STATUS
C             (2)  THE VEHICLE LOCATIONS
C             (3)  THE SIMULATION TIME
C
      COMMON/XLOCAL/ NTKLOC(20),NCRLOC(20),NCYLOC(20)
      INTEGER*2 FIELD,NTKLOC,NCRLOC,NCYLOC
      COMMON/TIMER/ ICLOCK,TIME,TNEXT
      COMMON/LIGHTS/  FIELD(160)
C
    1 FORMAT (1X,F8.3,72I1)
    2 FORMAT (1X,80I1)
    3 FORMAT (1X,20I4)
      WRITE(3,1) TIME,(FIELD(I),I = 9,80)
      WRITE(3,2) (FIELD(I),I = 81,160)
      WRITE(3,3) (NTKLOC(I),I = 1,7),(NCRLOC(I),I = 1,10),(NCYLOC(I),
     *I = 1,3)
      WRITE(3,3) (NTKLOC(I),I = 11,17),(NCRLOC(I),I = 11,20),(NCYLOC(I),
     *I = 11,13)
C
      RETURN
      END
```

APPENDIX B

CONTROL SYSTEM SOFTWARE CHANGE DOCUMENTATION

(NARRATIVES, FLOW CHARTS, LISTINGS)

MEMORY USAGE MAP

ADDRESS	PROGRAM
0 - 1F	INTERRUPTS *
20 - 3B6	EXECUTIVE *
3B7 - 48F	________________
490 - 536	MULTIPLY/DIVIDE *
537 - 5AC	________________
5AD - 7E9	OPERATOR MONITOR
7EA - 7FF	________________
800 - AE7	GREEN BAND UPDATE
AE8 - B7F	________________
B80 - FD4	HIGHWAY VEHICLE PROCESSOR *
FD5 - FFF	________________
1000 - 15C8	SITUATION DISPLAY *
15C9 - 15E1	________________
15E2 - 15FF	CHECK LIGHT IN BAND
1600 - 1776	DATA LOGGER *
1777 - 17FF	________________
1800 - 1F82	GREEN BAND STATUS
1F83 - 1FFF	________________
2000 - 2269	VELOCITY - VOLUME
227A - 23C7	RING BUFFER
23FB - 23FF	________________
2400 - 24B8	TABLE COPY *
24B9 - 25F0	BAND TRIM

* Note - This Routine or Data Table was not modified from its original state.

MEMORY USAGE (cont.)

25F1 - 27FF	————————
2800 - 2DF9	FAULT MONITOR
2DFA - 332F	————————
3330 - 390B	TAPE BUFFER ZERO*
390C - 3EE7	TAPE BUFFER ONE *
3EE8 - 3EFD	CHECK AND UPDATE SECOND COUNT
3EFE - 3EFF	————————
3F00 - 3FF4	LOAD AND DUMP
3FF5 - 3FFE	SENSOR CHECKOUT

*Note - This Routine or Data Table was not modified from its original state.

OPERATOR MONITOR SUBPROGRAM CHANGES

Three commands were added to OPERATOR MONITOR:

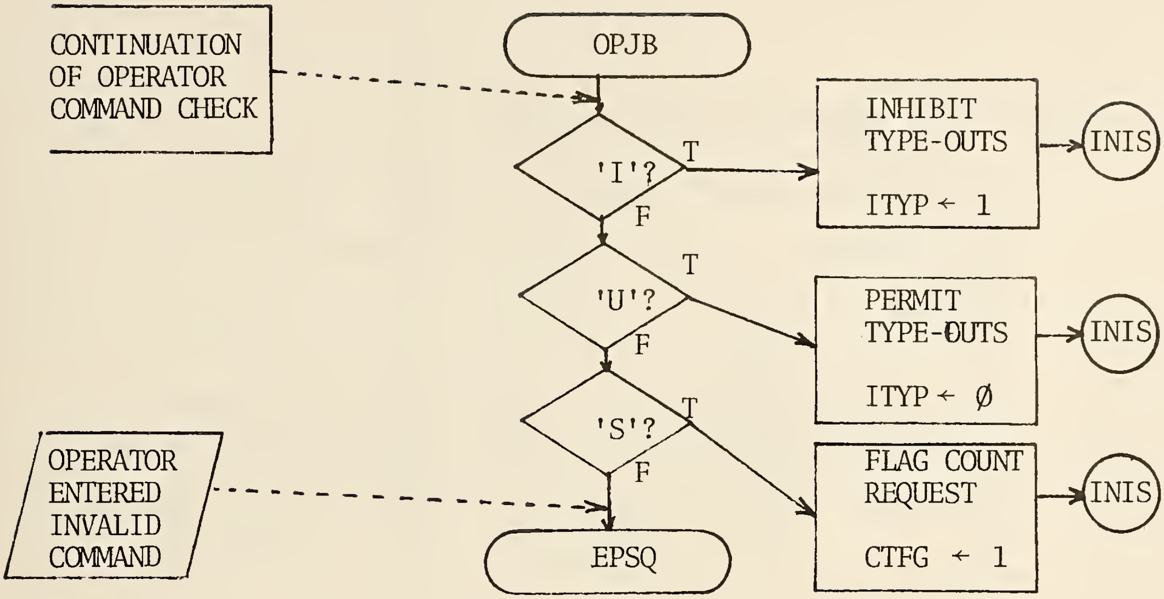
'I' Inhibit information type-outs,

'U' Uninhibit (permit) information type-outs and

'S' Print Sensor counts.

To implement the additions, control was intercepted following the last check in the command list, where it would transfer to the invalid input handling. A JMP command between labels OPLQ and OPMQ was changed to JMP OPJB to transfer to the added command checks.

At OPJB, at the front of the routine the command is checked for the additions and if one is not found, control transfers to the error handling, JMP EPSQ. For 'I', ITYP is set to 1 indicating type-outs are permitted; and for 'S', CTFG, count flag, is set to 1 (it is reset to Ø by routine doing type-out.)


CONTINUATION OF OPERATOR COMMAND CHECK
OPJB
'I'?
T
F
INHIBIT TYPE-OUTS
ITYP ← 1
INIS
'U'?
T
F
PERMIT TYPE-OUTS
ITYP ← Ø
INIS
'S'?
T
F
FLAG COUNT REQUEST
CTFG ← 1
INIS
OPERATOR ENTERED INVALID COMMAND
EPSQ

```
                'EXTENSION FOR OPERATOR COMMAND INPUT
                *
                *  ADDED 12/30/76  AHEAD OF PROGRAM
                *
     05AD       OPJB     EQU    $
05AD 07C9                CLB    'I'   INHIBIT TYPE-OUTS?
05AE 0860                SEQ
05AF 15B4                JMP    OPJC   NO, CHECK OTHER RESPONSES
05B0 87D0                LDW    =1    YES, SET FLAG
     05B1       OPSF     EQU    $
05B1 008A                SMB    ITYP        IN FAULT MONITOR
05B2 7003                STW    ITYP
05B3 17A0                JMP    INIS
                *
     05B4       OPJC     EQU    $
05B4 07D5                CLB    'U'   PERMIT TYPE-OUTS?
05B5 0860                SEQ
05B6 15B9                JMP    OPJD   NO, CONTINUE
05B7 0100                CLR
05B8 15B1                JMP    OPSF  RESET INHIBIT FLAG
                *
     05B9       OPJD     EQU    $
05B9 07D3                CLB    'S'  WANT SENSOR ACTIVATION COUNTS?
05BA 0860                SEQ
05BB 17B5                JMP    EPSQ  ** INVALID OPERATOR ENTRY **
05BC 87D0                LDW    =1   YES, FLAG FAULT MONITOR
05BD 008A                SMB    CTFG       FOR JOB
05BE 7004                STW    CTFG
05BF 17A0                JMP    INIS
```

```
                 ' OPERATOR MONITOR SUBPROGRAM REVISED
                 * VERSION OF 9-29-74
                 * OPERATOR INTERRUPT HANDLING ROUTINE
                             ORIG   X'5C0'
05C0 0031                    DSB    1
05C1 0100                    CLR
05C2 06CF                    LLB    X'CF'
05C3 7046                    STW    ERRCH
05C4 1020                    JMP    BEGT
05C5                         RES    X'05D5'-$
     0046         ERRCH      EQU    X'0046'
     0020         BEGT       EQU    X'0020'
05D5 0100         PCH2A      CLR
05D6 06C9                    LLB    X'C9'
05D7 7046                    STW    ERRCH
05D8 17A0                    JMP    INIS
05D9                         RES    2
                 * TAPE MESSAGE DATA RECEIVING ROUTINE
05DB 08F0         PCH1       SS3      IF DATA LOGGING NOT ON
05DC 15DE                    JMP    PCH1A    REJECT MESSAGES
05DD 17B5                    JMP    EPSQ
05DE 87CC         PCH1A      LDW    CHAQ
05DF F7D1                    CMW    =X'8D'      CR?
05E0 0870                    SNE
05E1 15E7                    JMP    PCH1B       YES, MESSAGE FINISHED
05E2 862B                    LDW    PN4Q
05E3 F045                    CMW    PN3T
05E4 0840                    SLS
05E5 15E7                    JMP    PCH1B    BUFFER FULL
05E6 1657                    JMP    OPAR     MORE ROOM IN BUFFER
05E7 87D2         PCH1B      LDW    =X'FFFF'
05E8 7629                    STW    FPNQ     SET BUFFER FULL FLAG
05E9 8629         PCH1C      LDW    FPNQ
05EA 0800                    SAZ
05EB 15E9                    JMP    PCH1C    WAIT TILL DATA LOGGER GRABS
05EC 17A0                    JMP    INIS     MESSAGE FROM BUFFER
05ED                         RES    X'05F8'-$
05F8              OPTS       RES    37         TTY INPUT BUFFER
061D                         RES    X'061E'-$
                 * EXEC LINKAGE
     0043         PN1Q   EQU X'0043'  BYTE BEGIN OF COMMAND BUFFER
     0027         OPIT   EQU X'0027'  KBD INTERRUPT LINK ADDRESS
     0028         OPPT   EQU X'0028'  PNT INTERRUPT LINK ADDRESS
     0044         PN2T   EQU X'0044'  FIRST EMPTY BYTE
     0045         PN3T   EQU X'0045'  LAST EMPTY BYTE+1
     0042         FL9T   EQU X'0042'  PRINT FLAG 0 WHEN PRINTING
     2801         FLMT   EQU X'2801'  FAULT MONITOR ENTRY
     2802         TMREQ  EQU X'2802'  TIME PRINT REQUEST FLAG
     2803         ITYP       EQU    X'2803'  INHIBIT TYPE-OUT FLAG
     2804         CTFG       EQU    X'2804'  PRINT SENSOR ACTIVATION FLAG
061E 0000         FPAQ    D    0 FLAG AND COUNTER FOR COMMA
061F 0000         FPBQ    D    0 FLAG AND COUNTER FOR T
0620 0000         FPCQ    D    0 FLAG AND COUNTER FOR
0621 0000         FPDQ    D    0 SET FOR WAITING-FOR-A-CHANGE
0622 0000         FPEQ    D    0
0623 0000         FPFQ    D    0 FLAG AND COUNTER FOR 1ST LOCATION
                 *                CHARACTERS (BEFORE T)
```

```
0624 0000     88 FPGQ  D    0 FLAG AND COUNTER FOR LAST LOCATION
              89 *            CHARACTERS (AFTER COMMA)
0625 0000     90 FPHQ  D    0 FLAG AND COUNTER FOR CHANGE CHARACTERS
0626 0000     91 FPJQ  D    0 DIGIT COUNTER .CLR AFTER EVERY 4
0627 0000     92 FPKQ  D    0 FLAG AND COUNTER FOR ;
0628 0000     93 FPMQ  D    0 TAPE MSG-OP MON INPUT FLAG
0629 0000     94 FPNQ  D    0 TAPE MSG FINISHED FLAG
062A 0000     95 FPQQ  D    0 ERROR FLAG
062B 0BF0     96 PN4Q  D    OPTS+OPTS LOCATION NEXT CHARACTER TO
              97 *              BE PROCESSED
062C 1638     98 OPMS  JMP OPAQ      OPERATOR MONITOR ENTRY LOOP
062D 008A     99 OPXW  SMB FLMT
062E 1001    100       JMP FLMT     EXIT TO FAULT MONITOR
062F 6637    101 INSS  STX TEMY      INITIALIZATION ENTRY
0630 87B4    102       LDW MODI
0631 77B2    103       STW RETZ      SET RETURN TO THIS CODING
0632 15D5    104       JMP   PCH2A
0633 87B3    105 INRT  LDW MODN      SET UP NORMAL RETURN
0634 77B2    106       STW RETZ
0635 9637    107       LDX TEMY
0636 1800    108       JMP * 0        INITIALIZATION RETURN
0637 0000    109 TEMY  DATA 0         SAVE RETURN INDEX
             110
```

```
               ' OPERATOR MONITOR SUBPROGRAM
0638 8044      OPAG  LDW PN2T
0639 F043            CMW PN1Q SOMETHING IN COMMAND BUFFER?
063A 0880            SGR
063B 16ED            JMP OPKW EXIT TO FAULT MONITOR
063C 03EB            DOT X'E',X'B'  ENABLE KEYBOARD  ***
063D 87D3            LDW =OPIT
063E 7001            STW 1
063F 9043            LDX PN1Q
0640 0050            SGM
0641 0100            CLR
0642 5800            LDB * 0 GET 1ST CHAR OF LINE
0643 0758            CLB 'X'   IS IT 'X'?
0644 0860            SEQ
0645 1649            JMP OPLQ  NO
0646 87DC            LDW =-1 SET FLAG FOR CICS
0647 7828            STW FPMQ
0648 1653            JMP OPMQ
0649 07D4      OPLQ  CLB 'T'  IS IT TIME REQ 'T'?
064A 0860            SEQ
064B 1650            JMP $+5
064C 87DD            LDW =1
064D 003A            SMB TMREQ
064E 7032            STW TMREQ
064F 17A0            JMP INIS
0650 07CD            CLB 'M'
0651 0860            SEQ       IF M THEN DON'T CHANGE FLAG
0652 15AD            JMP OPJB  CHECK ADDED RESPONSES
0653 962B      OPMQ  LDX PN4Q
0654 0401            IXS 1  BUMP BUFFER POINTER
0655 0000            HLT
0656 662B            STX PN4Q  END OF X OR M INPUT LOGIC ***
0657 03EB      OPAR  DOT X'E',X'B'
0658 87D3            LDW =OPIT
0659 7001            STW 1
065A 8044            LDW PN2T YES
065B F82B            CMW PN4Q IS THERE A CHARACTER TO PROCESS?
065C 0880            SGR
065D 1658            JMP OPAR+1    NO*********THAYER PATCH*********
065E 0100            CLR YES
065F 962B            LDX PN4Q
0660 0050            SGM
0661 5800            LDB * 0  OLD RAYTHEON PATCH
0662 77CC            STW CHAQ
0663 0401            IXS 1
0664 0000            HLT
0665 662B            STX PN4Q BUMP BYTE POINTER
0666 8628            LDW FPMQ
0667 0820            SAM
0668 15DB            JMP PCH1
0669 87CC            LDW CHAQ
066A 07AF      OPAS  CLB '/' IS CHARACTER A HEX NUMBER?
066B 0880            SGR
066C 1725            JMP OPBQ MAY BE A LEGAL NON-NUMBER
066D 0739            CLB '9'
066E 0880            SGR
066F 167D            JMP OPBR HEX NO. B0-B9
```

```
0670 07DA            CLB ':'
0671 0C80            SGR
0672 17B5            JMP EPSQ ILLEGAL CHARACTER
0673 07DB            CLB ';'
0674 0C80            SGR
0675 1725            JMP OPBQ CHARACTER IS BB
0676 07C0            CLB '@'
0677 0C80            SGR
0678 17B5            JMP EPSQ ILLEGAL CHARACTER
0679 07C6            CLB 'F'
067A 0C80            SGR
067B 1680            JMP OPBS HEX NO. C1-C6
067C 1725            JMP OPBQ MAY BE A LEGAL NON-NUMBER
067D B7D4       OPBR SUB ='0'
067E 77CC            STW CHAQ
067F 1683            JMP OPCQ
0680 B7D5       OPBS SUB ='A'
0681 A7D6            ADD =10
0682 77CC            STW CHAQ
```



```
               ' OPERATOR MONITOR SUBPROGRAM
0683 861F       OPCQ LDW FPSQ
0684 F7D0            CMW =1 COMMAND COMPLETE(SEEN A T)?
0685 0860            SEQ
0686 16FF            JMP OPTQ NO
0687 8C21            LDW FPDQ YES
0688 F7D0       OPCS CMW =1 WAITING FOR A CHANGE?
0689 0860            SEQ
068A 169B            JMP OPFQ NO
068B 8625            LDW FPHQ
068C A7D0            ADD =1
068D 7625            STW FPHQ
068E F7D7            CMW =5 MORE THAN 4 CHG CHARACTERS?
068F 0840            SLS
0690 17B5            JMP EPSQ YES-ERROR
0691 F7D0       OPES CMW =1 FIRST CHANGE NUMBER?
0692 0860            SEQ
0693 1698            JMP OPDQ NO
0694 0100            CLR      YES
0695 A7CC       OPEQ ADD CHAQ
0696 77C9            STW NUCQ
0697 1658            JMP OPAR+1
0698 87C9       OPDQ LDW NUCQ
0699 0914            SLA 4
069A 1695            JMP OPEQ
069B 8044       OPFQ LDW PN2T
069C F62B            CMW PN4Q  EXTRA CHARACTER IN COMMAND BUFFER?
069D 0860            SEQ
069E 17B5            JMP EPSQ YES-ERROR
069F 861E            LDW FPAQ NO
06A0 F7D0            CMW =1 MULTIPLE PRINT-OUT?
06A1 0860            SEQ
06A2 16A4            JMP $+2
06A3 16A6            JMP OPFS
06A4 97D8       OPDR LDX =OPQQ
06A5 16B9            JMP OPOQ
```

```
                ' OPERATOR MONITOR SUBPROGRAM
06A6 97D9       OPFS  LDX =OPFR
06A7 178A             JMP OQCQ
06A8 87C7       OPFR  LDW NUAQ
06A9 26E0             JSX OPQR PRINT START LOC OR 1ST LOC ON THE LIN
06AA 26B9             JSX OPOQ PRINT 2 SPACES
06AB 97DA       OPHQ  LDX =OPGQ
06AC 16DA             JMP OPQQ PRINT 1ST OR NEXT "CONTENTS" WORD
06AD 26B9       OPGQ  JSX OPOQ
06AE 87C7             LDW NUAQ
06AF F7C8             CMW NUBQ FINISHED?
06B0 0840             SLS    LAST WORD PRINTED?
06B1 17A0             JMP INIS YES
06B2 A7D0             ADD =1 NO
06B3 77C7             STW NUAQ INCREMENT INSPECT LOCATION
06B4 E7DB             AND =7 ISOLATE BITS 13,14,15
06B5 0800             SAZ       NEXT LOCATION END IN 0 OR 8?
06B6 16AA             JMP OPHQ-1 NO
06B7 278A             JSX OQCQ YES. PRINT RET-RUB-LF
06B8 16A8             JMP OPFR
```

```
                ' OPERATOR MONITOR SUBPROGRAM
06B9 67CB       OPOQ  STX NUAS
06BA 87DC             LDW =X'A0' SUBROUTINE TO PRINT 2 SPACES
06BB 77CE             STW ASCR
06BC 26C8             JSX OPJQ
06BD 87DC             LDW =X'A0'
06BE 77CE             STW ASCR
06BF 26C8             JSX OPJQ
06C0 97CB             LDX NUAS
06C1 1800             JMP * 0
06C2 0100       OPRQ  CLR
06C3 7626             STW FPJQ CLR DIGIT COUNTER
06C4 87D0             LDW =1
06C5 7621             STW FPDQ SET WAITING-FOR-A-CHANGE
06C6 03EB             DOT X'E',X'B' ENABLE KEYBOARD
06C7 1657             JMP OPAR LOOK FOR ANOTHER CHARACTER
06C8 87DD       OPJQ  LDW =OPPT
06C9 7001             STW 1
06CA 87D0             LDW =1
06CB 7042             STW FLST
06CC 87CE             LDW ASCR
06CD 03EA             DOT X'E',X'A' TURN ON PRINTER
06CE 03EE             DOT X'E',X'E' OUTPUT ACR TO PRINTER
06CF 02E0             DIN X'E',X'0'  STATUS CHECK
06D0 0A17             SLL 7
06D1 0820             SAM
06D2 16D6             JMP $+4
06D3 0100             CLR
06D4 7042             STW FLST
06D5 03E0             DOT X'E',X'0'
06D6 8042             LDW FLST LOOK AT FLAG
06D7 0800             SAZ SKIP IF CLR - PRINTING STOPPED
06D8 16CF             JMP $-9 WAIT SEQUENCE
06D9 1800             JMP * 0
06DA 67CB       OPQQ  STX NUAS
06DB 0050             SGM
06DC 97C7             LDX NUAQ
06DD 8800             LDW * 0 CONTENTS OF CONTENTS OF NUAQ
06DE 0040             SLM
06DF 16E1             JMP $+2
06E0 67CB       OPQR  STX NUAS
06E1 77CD             STW OPKS
06E2 97CD       OPQS  LDX OPKS
06E3 0100             CLR
06E4 0A74             SLC D 4
06E5 A7D4             ADD ='0'
06E6 07B9             CLB '9'
06E7 0880             SGR
06E8 16EA             JMP $+2 DIGIT IS 0-9
06E9 A7DB             ADD =7 DIGIT IS A-F
06EA 77CE             STW ASCR
06EB 67CD             STX OPKS SAVE DIGITS TO BE PRINTED
06EC 26C8             JSX OPJQ
06ED 8626             LDW FPJQ
06EE A7D0             ADD =1
06EF 7626             STW FPJQ
```

```
               ' OPERATOR MONITOR SUBPROGRAM
06F0 F7DE            CMW =4 FINISHED PRINTING?
06F1 0860            SEQ
06F2 16E2            JMP OPQS NO-CONTINUE DIGIT ENCODING & PRINTOUT
06F3 8627            LDW FPKQ ; FLAG
06F4 0800            SAZ "LOCATION" GROUP JUST PRINTED?(; FLAG SET?
06F5 179A            JMP OQCR YES
06F6 861E            LDW FPAQ NO
06F7 0800            SAZ
06F8 16FB            JMP $+3 A COMMA HAS BEEN SEEN
06F9 97DF            LDX =OPRQ COMMA HAS NOT BEEN SEEN
06FA 16B9            JMP OPOQ
06FB 0100            CLR
06FC 7626            STW FPJQ CLEAR DIGIT PRINTOUT CTR
06FD 97C3            LDX NUAS
06FE 1300            JMP * 0
06FF 861E      OPTQ  LDW FPAQ
0700 F7D0            CMW =1  MULTIPLE PRINTOUT?(SEEN A COMMA)
0701 0860            SEQ
0702 1715            JMP OPWQ NO
0703 8624            LDW FPGQ YES
0704 A7D0            ADD =1
0705 7624            STW FPGQ
0706 F7DE            CMW =4 MORE THAN 4 CHARAC TO RIGHT OF COMMA?
0707 0880            SGR
0708 170A            JMP $+2 NO-OK
0709 17B5            JMP EPSQ YES-ERROR
070A 8624            LDW FPGQ
070B F7D0            CMW =1 FIRST HEX DIGIT?
070C 0860            SEQ
070D 1712            JMP OPVQ NO
070E 0100            CLR YES
070F A7CC      OPUQ  ADD CHAQ
0710 77C8            STW NUBQ
0711 1657            JMP OPAR
0712 87C8      OPVQ  LDW NUBQ
0713 0914            SLA 4
0714 170F            JMP OPUQ
0715 8623      OPWQ  LDW FPFQ
0716 A7D0            ADD =1
0717 7623            STW FPFQ
0718 F7D7            CMW =5 MORE THAN 4 CHARACTERS?
0719 0840            SLS
071A 17B5            JMP EPSQ YES-ERROR
071B F7D0            CMW =1 NO.FIRST HEX DIGIT?
071C 0860            SEQ
071D 1722            JMP OPYQ NO
071E 0100            CLR YES
071F A7CC      OPXQ  ADD CHAQ ACCUMULATE HEX DIGIT
0720 77C7            STW NUAQ
0721 1657            JMP OPAR
0722 87C7      OPYQ  LDW NUAQ
0723 0914            SLA 4
0724 171F            JMP OPXQ
```

```
                ' OPERATOR MONITOR SUBPROGRAM
0725 87CC  OPBQ   LDW CHAQ
0726 07AC         CLB ',' IS IT A COMMA?
0727 0860         SEQ
0728 173A         JMP OPZQ NO
0729 861E         LDW FPAQ YES
072A A7D0         ADD =1
072B 761E         STW FPAQ
072C F7E0         CMW =2 MORE THAN 1 COMMA?
072D 0840         SLS
072E 17B5         JMP EPSQ YES
072F 8623         LDW FPFQ NO
0730 0800         SAZ HAS A HEX NUMBER BEEN SEEN?
0731 1733         JMP $+2
0732 17B5         JMP EPSQ HAVEN'T SEEN HEX- ERROR
0733 861F         LDW FPBQ
0734 0800         SAZ SEEN A T?
0735 17B5         JMP EPSQ YES-ERROR
0736 8627         LDW FPKQ
0737 0800         SAZ       SEEN A ;?
0738 17B5         JMP EPSQ  YES - ERROR
0739 1657         JMP OPAR
073A 87CC  OPZQ   LDW CHAQ
073B 07D4         CLB 'T' IS IT A T?
073C 0860         SEQ
073D 174F         JMP OQAQ NO
073E 861F         LDW FPBQ YES
073F A7D0         ADD =1
0740 761F         STW FPBQ
0741 8623         LDW FPFQ
0742 0800         SAZ SEEN A HEX?
0743 1745         JMP $+2 YES
0744 17B5         JMP EPSQ NO-ERROR
0745 03E0         DOT X'E',X'0'
0746 861E         LDW FPAQ
0747 0800         SAZ SEEN A COMMA?
0748 174A         JMP $+2 YES
0749 1683         JMP OPCQ NO
074A 8624         LDW FPGQ
074B 0800         SAZ IS THERE A HEX NO. AFTER THE COMMA?
074C 174E         JMP $+2 YES
074D 17B5         JMP EPSQ NO
074E 1683         JMP OPCQ
```

```
                ' OPERATOR MONITOR SUBPROGRAM
074F 87CC  OQAQ  LDW CHAQ
0750 07AE        CLB '.' IS IT A PERIOD?
0751 0860        SEQ
0752 1774        JMP OQAS NO
0753 03E0        DOT X'E',X'0' YES
0754 8620        LDW FPCQ
0755 A7D0        ADD =1
0756 7620        STW FPCQ
0757 861E  OQBS  LDW FPAQ
0758 0800        SAZ SEEN A COMMA?
0759 17B5        JMP EPSQ YES-ERROR
075A 861F        LDW FPBQ NO
075B 0800        SAZ SEEN A T?
075C 175E        JMP $+2 YES
075D 17B5        JMP EPSQ
075E 8621        LDW FPDQ
075F 0830        SAO WAITING-FOR-CHG FLAG SET ?
0760 17B5        JMP EPSQ NO-ERROR
0761 8044        LDW PN2T
0762 F62B        CMW PN4Q EXTRA CHARACTER IN COMMAND BUFFER?
0763 0880        SGR
0764 1766        JMP $+2 NO
0765 17B5        JMP EPSQ YES
0766 8620        LDW FPCQ SEEN A . ?
0767 0800        SAZ
0768 176A        JMP $+2 YES
0769 177D        JMP OQBR NO. RETURN TO ; PROCESSING
076A 8625        LDW FPHQ
076B 0800        SAZ OPERATOR WISHES TO MAKE A CHANGE ?
076C 176E        JMP $+2 YES
076D 17A0  OQAR  JMP INIS NO
076E 87C9        LDW NUCQ
076F 0050        SGM
0770 97C7        LDX NUAQ
0771 7800        STW * 0
0772 0040        SLM
0773 17A0        JMP INIS
```

```
               ' OPERATOR MONITOR SUBPROGRAM
0774 87CC      OQAS  LDW CHAQ
0775 07BE            CLB ';' IS IT A ;?
0776 0860            SEQ
0777 17B5            JMP EPSQ NO- ILLEGAL CHARACTER
0778 03E0            DOT X'E',X'0' TURN OFF TT
0779 8627            LDW FPKQ
077A A7D0            ADD =1
077B 7627            STW FPKQ
077C 1757            JMP OQBS
077D 8625      OQBR  LDW FPHQ
077E 0800            SAZ SOMETHING IN CONTENTS-CHG BUFFER?
077F 1781            JMP $+2 YES
0780 1789            JMP PQCQ NO
0781 87C9            LDW NUCQ
0782 0050            SGM
0783 97C7            LDX NUAQ
0784 7800            STW * 0
0785 0040            SLM
0786 0100            CLR
0787 7625            STW FPHQ
0788 7621            STW FPDQ
0789 97E1      PQCQ  LDX =OQCS
078A 67CB      OQCQ  STX NUAS
078B 87D1            LDW =X'8D' RETURN-RUBOUT-LF
078C 77CE            STW ASCR
078D 26C8            JSX OPJQ
078E 87E2            LDW =X'FF'
078F 77CE            STW ASCR
0790 26C8            JSX OPJQ
0791 87E3            LDW =X'8A'
0792 77CE            STW ASCR
0793 26C8            JSX OPJQ
0794 97CB            LDX NUAS
0795 1800            JMP * 0
```

```
               477 ' OPERATOR MONITOR SUBPROGRAM
0796 87C7      478 OQCS  LDW NUAQ
0797 A7D0      479       ADD =1
0798 77C7      480       STW NUAQ
0799 16E0      481       JMP OPQR
079A 97D8      482 OQCR  LDX =OPQQ
079B 67CB      483 PQCR  STX NUAS
079C 0100      484       CLR
079D 7627      485       STW FPKQ CLEAR ; FLAG
079E 7626      486       STW FPJQ CLEAR DIGIT COUNTER
079F 16E9      487       JMP OPOQ
07A0 97E4      488 INIS  LDX =OQDQ
07A1 178A      489       JMP OQCQ
07A2 03E0      490 OQDQ  DOT X'E',0
07A3 97E5      491       LDX =35
07A4 0100      492       CLR
07A5 7DF8      493 OQEQ  STW * OPTS CLEAR BUFFER AND FLAGS
07A6 0501      494       DXS 1
07A7 17A5      495       JMP OQEQ
07A8 97E6      496       LDX =11
07A9 7E1E      497 OQGQ  STW * FPAQ
07AA 0501      498       DXS 1
07AB 17A9      499       JMP OQGQ
07AC 8043      500       LDW PN1Q INITIALIZE POINTERS
07AD 7044      501       STW PN2T
07AE 762B      502       STW PN4Q
07AF 87D3      503       LDW =OPIT
07B0 7001      504       STW 1       SET KBD LINKAGE
07B1 03EB      505       DOT X'E',X'B'
07B2 0000      506 RETZ  DATA 0   MODIFIABLE RETURN INSTRUCTION
07B3 1633      507 MODN  JMP OPAQ  NORMAL RETURN INSTRUCTION
07B4 1633      508 MODI  JMP INRT  INITIALIZATION RETURN INSTRUCTION
               509
```

```
                 ' OPERATOR MONITOR SUBPROGAAM
07B5 03E0        EPSQ  DOT X'E',0
07B6 03EA              DOT X'E',X'A' ENABLE PRINTER
07B7 06BF        OQFQ  LLB '?' QUESTION MARK
07B8 77CE              STW ASCR
07B9 26C8              JSX OPJQ
07BA 06AA              LLB '*'
07BB 77CE              STW ASCR
07BC 26C8              JSX OPJQ
07BD 06A1              LLB '!'
07BE 77CE              STW ASCR
07BF 26C8              JSX OPJQ
07C0 06BF              LLB '?'
07C1 77CE              STW ASCR
07C2 26C8              JSX OPJQ
07C3 87D2              LDW =-1
07C4 762A              STW FPQQ ERROR FLAG SET
07C5 278A              JSX OQCQ PRINT RET-RUBOUT-LF
07C6 17A0        OQFR  JMP INIS
07C7 0000        NUAQ  D    0 LOC OF 1ST (OR NEXT) INSPECT LOC
07C8 0000        NUBQ  D    0 LOC OF LAST INSPECT(AND PRINTOUT) LOC
07C9 0000        NUCQ  D    0 LOC OF VALUE OF ACCUMULATED CHG
07CA 0000        NUAR  D    0
07CB 0000        NUAS  D    0 IXR SAVER
07CC 0000        CHAR  D    0 CHARACTER IN PROCESS
07CD 0000        OPKS  D    0 PRINT-OUT DIGIT SAVER
07CE 0000        ASCR  D    0 STORE ASCII
07CF 0000        ACRR  D    0 STORE ACR
                 *
                       END

07D0 0001
07D1 00ED
07D2 FFFF
07D3 0027
07D4 00B0
07D5 00C1
07D6 000A
07D7 0005
07D8 06DA
07D9 06A8
07DA 06AD
07DB 0007
07DC 00A0
07DD 0026
07DE 0004
07DF 06C2
07E0 0002
07E1 0796
07E2 00FF
07E3 008A
07E4 07A2
07E5 0023
07E6 000B
07E7 0000
07E8 0000
07E9 0000            NO ERRORS
```

ACRR	07CF	ASCR	07CE	DIGT	0030	CHAR	07CC
CTRC	2904	EPSQ	07D5	ERRCK	0046	FLGT	0042
FLMT	2801	FPAQ	061E	FPBQ	061F	FPCQ	0620
FPDQ	0621	FPEQ	0622	FPFQ	0623	FPGQ	0624
FPHQ	0625	FPJQ	0626	FPKQ	0627	FPMQ	0628
FPNQ	0629	FPOQ	062A	INIC	07A0	INPT	0633
INSC	062F	ITYP	2808	MODI	07D4	MODM	07D3
NUAQ	07C7	NUAR	07C4	NUAS	07CE	NUBQ	07C8
NUCQ	07C0	OPAQ	0656	OPAR	0657	OPAS	066A
OPBQ	0725	OPBR	067D	OPBS	0680	OPCQ	0683
OPCS	068E	OPDQ	0693	OPDS	0644	OPEQ	0695
OPES	0691	OPFQ	069B	OPFR	06A8	OPFS	06A6
OPGQ	05AD	OPHQ	06AB	OPIT	0037	OPJB	05AE
OPJC	05B4	OPJD	0559	OPJQ	06C6	OPKS	07CE
OPLQ	0649	OPMQ	0653	OPMS	06CC	OPOQ	06D9
OPPT	0023	OPQQ	06DA	OPQR	06E0	OPQS	06E2
OPRQ	06C2	OPSF	05B1	OPTQ	06FF	OPTS	05FC
OPUQ	070F	OPVQ	071E	OPWQ	0715	OPXQ	071F
OPXR	062D	OPYQ	0722	OPZQ	073A	OQAQ	074F
OQAR	076D	OQAS	0774	OQBQ	077D	OQBS	0757
OQCQ	073A	OQCR	079A	OQCS	0796	OQDQ	07A2
OQEQ	07A5	OQFQ	07D7	OQFR	0706	OQGQ	07A3
PCH1	05DB	PCH1A	05DE	PCH1B	05E7	PCH1C	05E9
PCH2A	05D5	PN1Q	0043	PN2T	0044	PN3T	0045
PN4Q	062B	PQCQ	0789	PQCR	079B	RETN	07D?
TEMY	0637	TMRET	2802				

PASS?

GREEN BAND UPDATE SUBPROGRAM

The changes to the routine are:

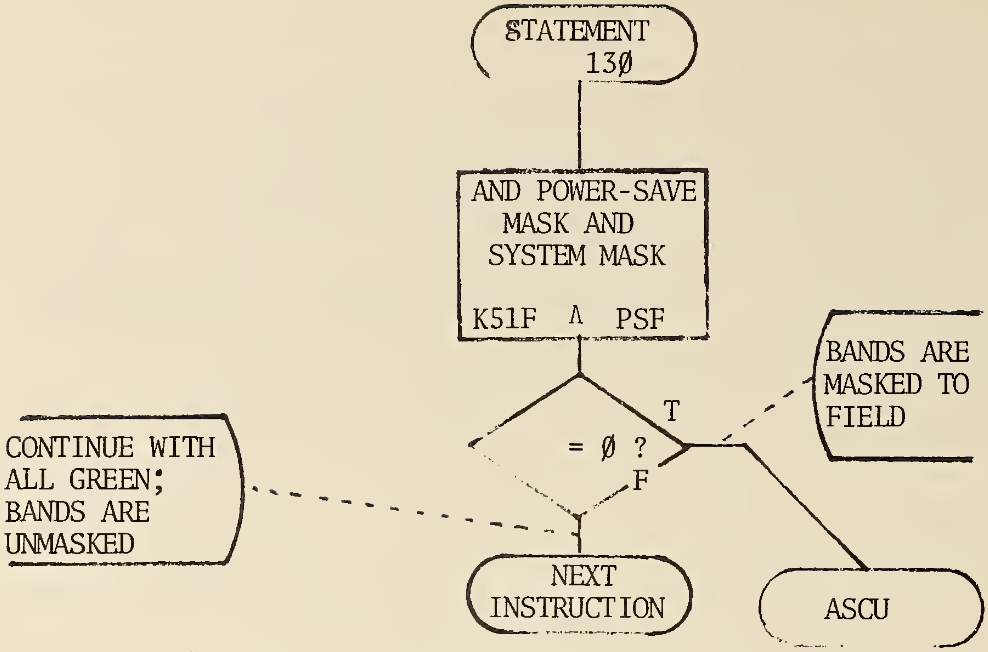
1. Inclusion of a flag for power-save masking of green band, which will not reflect in logged data or change the overhead sign;
2. Remove separate calculation of bands for Situation Display and replace it with an abbreviated, or condensed, output of what is output to the field display, to reduce computation time; and
3. Insert call to BANDTRIM routine which will smooth the display before presentation to drivers on the ramp.

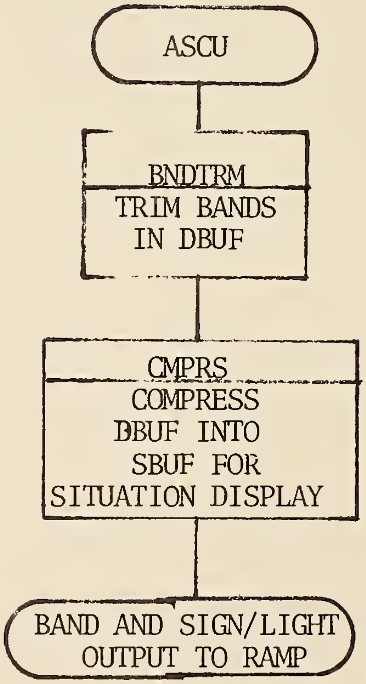
Two external routines were added; CUSC, a routine to provide a second count over the range 0 - 32767, and BNDTRM, the band trimming routine. PSF, the power-save flag is an added external data reference.

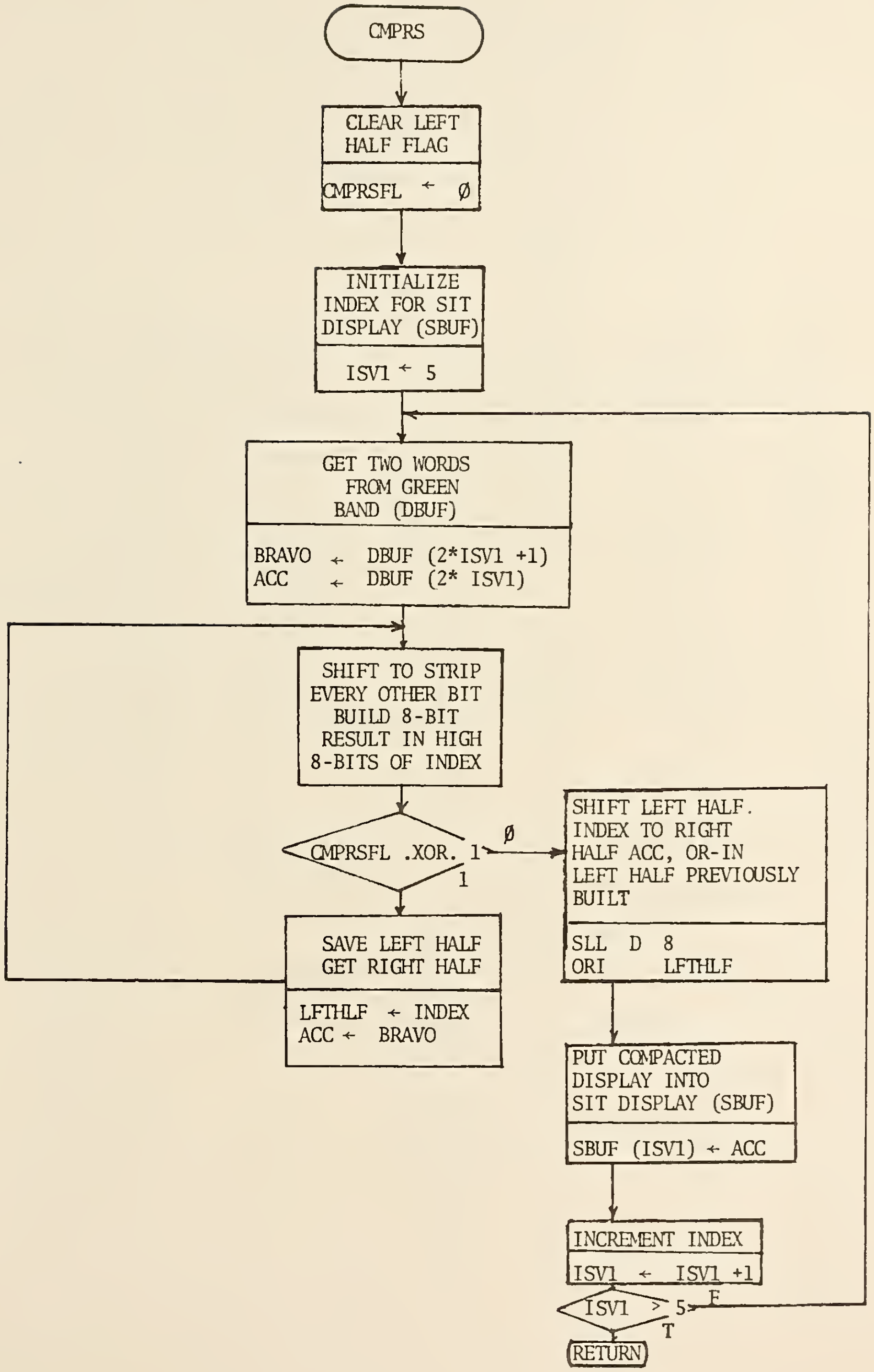
Following the main entry, BGUD, a power-save mask (PSF) check was included with a band-mask (KSIF) check. They must both indicate "unmasked" for the system to proceed.

Before label CHEK, the routine to calculate bands for the Situation Display was removed and a call to subroutine CMPRS added following the BNDTRM (Band Trim) to fill Situation Display Buffer (SBUF). CMPRS strips every other list (one bit per lamp) from the Green Band Display Buffer (DBUF) and puts the result into SBUF. (10 words are packed into 5.)

At the end of CKPD, check if merge area has been occupied for 5 seconds, sensor R11 is checked. It controls the power-save mask. When sensor R11 has gone unactivated for 30 seconds, PSF (power-save mask) is set to mask the bands but not change the overhead sign until R11 is activated.


STATEMENT 13Ø
AND POWER-SAVE MASK AND SYSTEM MASK
K51F ∧ PSF
= Ø ?
T
F
BANDS ARE MASKED TO FIELD
CONTINUE WITH ALL GREEN; BANDS ARE UNMASKED
NEXT INSTRUCTION
ASCU


ASCU
BNDTRM
TRIM BANDS IN DBUF
CMPRS
COMPRESS DBUF INTO SBUF FOR SITUATION DISPLAY
BAND AND SIGN/LIGHT OUTPUT TO RAMP



CMPRS
CLEAR LEFT HALF FLAG
CMPRSFL ← Ø
INITIALIZE INDEX FOR SIT DISPLAY (SBUF)
ISV1 ← 5
GET TWO WORDS FROM GREEN BAND (DBUF)
BRAVO ← DBUF (2*ISV1 +1)
ACC ← DBUF (2* ISV1)
SHIFT TO STRIP EVERY OTHER BIT BUILD 8-BIT RESULT IN HIGH 8-BITS OF INDEX
CMPRSFL .XOR. 1
Ø
1
SAVE LEFT HALF GET RIGHT HALF
LFTHLF ← INDEX
ACC ← BRAVO
SHIFT LEFT HALF. INDEX TO RIGHT HALF ACC, OR-IN LEFT HALF PREVIOUSLY BUILT
SLL D 8
ORI LFTHLF
PUT COMPACTED DISPLAY INTO SIT DISPLAY (SBUF)
SBUF (ISV1) ← ACC
INCREMENT INDEX
ISV1 ← ISV1 +1
ISV1 > 5
F
T
RETURN

```
                * GREEN BAND UPDATE SUBPROGRAM #1
                * VERSION PREPARED 10-13-74
                *   MODIFIED 12-2-76
                * THIS ASSY HAS VAR ACC BANDS IN SG
                * THIS ASSY HAS CONSTANT 3 FT/SEC**2 ACC IN MM
                * THIS ASSY SERVICES A 608 FT RAMP
                *
                * PROCEDURE TO BUILD OUTPUT TO SCU
                *
                DB11      PROC
                          LDW   DBUF+P(1)-1
                          DOT   0,P(1)
                          ENDP
                *
                          ORIG  X'800'    ABSO ASSY START
      0020      INT       EQU   X'20'     EXECUTIVE INIT RETURN ADDRES
      0021      TIME      EQU   X'21'     2 MSEC CLOCK
      0025      EXEK      EQU   X'25'     EXECUTIVE RETURN ADDRESS
      1802      MODE      EQU   X'1802'   SYSTEM MODE
      1804      INFG      EQU   X'1804'   INITIALIZATION FLAG
      1805      VMRG      EQU   X'1805'   GREEN BAND SPEED
      10C8      SBUF      EQU   X'10C8'   SIT DISPLAY OUTPUT BUFFER
      180C      K51F      EQU   X'180C'   FLAG TO MASK GREEN BAND
      180D      PSF       EQU   X'180D'   POWER SAVE MASK
      247A      WORKFLG   EQU   X'247A'   TABLE COPY IN PROGRESS FLAG
      0170      SPPS      EQU   X'170'    PRESENCE DETECTOR INDICATOR
      0178      TPPS      EQU   X'178'    TIME P.D. LAST ACTIVATED
      0490      SPMUL     EQU   X'490'    SOFTWARE MULTIPLY
      24B9      BNDTRM    EQU   X'24B9'       BANDTRIM ROUTINE
      24BA      LST       EQU   X'24BA'   TIME SAVE IN BNDTRM
      3EE8      CUSC      EQU   X'3EE8'   CHECK SECOND COUNT ROUTINE
0800            LAMJ      RES   40    OLD PACER LIGHT DISPLAY BUFFER
0828 0000                 D     0
0829            DBUF      RES   11    DRIVER DISPLAY OUTPUT BUFFER
0834 10F8                 JMP   GBUD
0835 10E0                 JMP   INIT
0836 0000       NBND      D     0    NUMBER OF BANDS IN TABLE
0837            GBT       RES   120   GREEN BAND TABLE
08AF 0000       ALLG      D     0   ALL GREEN FLAG
                *
                * TABLE USED EXTERNALLY
                *
08B0 FFFF       TMSK      D     X'FFFF'    MASK TABLE
08B1 7FFF                 D     X'7FFF'
08B2 3FFF                 D     X'3FFF'
08B3 1FFF                 D     X'1FFF'
08B4 0FFF                 D     X'FFF'
08B5 07FF                 D     X'7FF'
08B6 03FF                 D     X'3FF'
08B7 01FF                 D     X'1FF'
08B8 00FF                 D     X'FF'
08B9 007F                 D     X'7F'
08BA 003F                 D     X'3F'
08BB 001F                 D     X'1F'
08BC 000F                 D     X'F'
08BD 0007                 D     X'7'
08BE 0003                 D     X'3'
```

```
         GREEN BAND UPDATE SUBPROGRAM  #1                          PAGE   2

08BF 0001      58              D        X'1'
08C0 0000      59              D        X'0'
               60



         GREEN BAND UPDATE SUBPROGRAM  #2                          PAGE   3

               61 '  GREEN BAND UPDATE SUBPROGRAM  #2
               62 *
08C1 0064      63 P100         D        100
08C2 09C4      64 PDMAX        D        X'9C4'   MAX P.D. ON TIME (5 SEC)
08C3 0004      65 ACCTIME      D        4   CONSTANT ACCELERATION IN FEET
08C4 0000      66 ACRT         D        0   TEMPORARY STORAGE
08C5 0000      67 AT           D        0   TEMPORARY STORAGE
08C6 0000      68 BKFL         D        0   BREAKDOWN FLAG
08C7 0000      69 ART1         D        0
08C8 0000      70 BCNT         D        0   PROCESSED BAND COUNTER
08C9 0000      71 COMF         D        0   COMPRESS TABLE FLAG
08CA 0000      72 DBFSBFLG D            0   DBUF-SBUF FLAG
08CB 0000      73 ART2         D        0
08CC 0000      74 DX           D        0
08CD 0000      75 FRAC         D        0   TEMPORARY STORAGE
08CE 0000      76 INTG         D        0   TEMPORARY STORAGE
08CF 0000      77 IXRT         D        0   TEMPORARY STORAGE
08D0 0000      78 MAXF         D        0
08D1 0000      79 NWRDFILL D            0
08D2 0000      80 OLDV         D        0
08D3 0000      81 SAVE         D        0   TEMPORARY STORAGE
08D4 0000      82 ART3         D        0
08D5 0000      83 TEM          D        0   TEMPORARY STORAGE
08D6 0000      84 TIX          D        0   TABLE INDEX
08D7 0000      85 TIXE         D        0   TABLE INDEX - POINTS TO XLE OR XE
08D8 0000      86 VAT          D        0   VAT = V(0) + 1/2AT
08D9 0000      87 XL           D        0   LEADING EDGE IN FEET
08DA 0000      88 XT           D        0   TRAILING EDGE IN FEET
08DB 0000      89 XLB          D        0   LEADING BIT NUMBER
08DC 0000      90 XTB          D        0   TRAILING BIT NUMBER
08DD 0000      91 XLW          D        0   LEADING WORD NUMBER
08DE 0000      92 XTW          D        0   TRAILING WORD NUMBER
               93
```

```
                ' GBUD INIT ROUTINE  #3
08DF 0000
08E0 60DF   INIT    SUBR
08E1 0100           CLR
08E2 0089           SMB   WORKFLG
08E3 747A           STW   WORKFLG   INIT TABLE COPY FLAG
08E4 0089           SMB   LST  TIME SAVE IN BAND TRIM
08E5 74BA           STW   LST
08E6 7036           STW   NBND    CLEAR NUMBER OF BANDS IN TABLE
08E7 82CF           LDW   =1      SET ALL GREEN FLAG-NOT TO SET
08E8 70AF           STW   ALLG    ENTIRE RAMP GREEN
08E9 0100           CLR
                    DO    1,11    CLEAR 11 SCU WORDS
08EA 0301           DOT   0,?
08EB 0302
08EC 0303
08ED 0304
08EE 0305
08EF 0306
08F0 0307
08F1 0308
08F2 0309
08F3 030A
08F4 030B
08F5 030F           DOT   0,X'F'   DUMMY SCU BOX STROBE
08F6 90DF           EXIT  INIT  RETURN
08F7 2800
```

```
                ' GBUD.MAIN ENTRY POINT  #4
08F8 0040         GBUD     SLM
08F9 0089                  SMB    WORKFLG
08FA 847A                  LDW    WORKFLG  DON'T DO GBUD WHILE
08FB 0800                  SAZ                TABLE COPY IS AT IT
08FC 11E1                  JMP    ASCU
08FD 0100                  CLR
08FE 92D0                  LDX    =-10   CLEAR DRIVER DISPLAY AND SIT
08FF 7833                  STW  * DBUF+10   DISPLAY OUTPUT BUFFERS
0900 0401                  IXS    1
0901 10FF                  JMP    $-2
0902 0086                  SMB    INFG
0903 8004                  LDW    INFG   INITIALIZING?
0904 0810                  SAP
0905 11E1                  JMP    ASCU   YES
0906 2257                  JSX    CKPD   NO. CHECK 8 RAMP P.D.S
0907 0040                  SLM
0908 80AF                  LDW    ALLG    IF=0, NO VEH ON HWY NOW.
0909 0800                  SAZ
090A 1118                  JMP    SETC   YES
090B 0086                  SMB    K51F
090C 800C                  LDW    K51F   CHECK K51F TO SEE IF
090D 0086                  SMB    PSF  ALSO POWER SAVE MASK,
090E E00D                  AND    PSF
090F 0800                  SAZ          BANDS ARE TO BE MASKED
0910 1112                  JMP    $+2
0911 11E1                  JMP    ASCU   IF SO SEND CLEARED DISPLAY
0912 92D0                  LDX    =-10   NO. SET ENTIRE DD GREEN
0913 82D1                  LDW    =-1
0914 7833                  STW  * DBUF+10
0915 0401                  IXS    1
0916 1114                  JMP    $-2
0917 11E1                  JMP    ASCU   OUTPUT DBUF TO DD
0918 0100         SETC     CLR        CLEAR
0919 7008                  STW    BCNT   PROCESSED BAND COUNTER
091A 70C9                  STW    COMF   COMPRESS TABLE FLAG
091B 8036                  LDW    NBND   ANY BANDS IN TABLE
091C 0800                  SAZ
091D 111F                  JMP    $+2
091E 11E1                  JMP    ASCU   NO.
091F 0A13                  SLL    3   YES
0920 B2D2                  SUB    =8
0921 70C7                  STW    ART1
0922 0100                  CLR
0923 70D6                  STW    TIX
0924 0080                  SMB    TIME
0925 8021                  LDW    TIME
0926 70D4                  STW    ART3
0927 112F                  JMP    PRCBAND
```

```
               ' GREEN BAND UPDATE SUBPROGRAM #5
0928 80C8        BEGI    LDW   BCNT
0929 F036                CMW   NBND   PROCESSED ALL BANDS?
092A 0870                SNE
092B 11CD                JMP   FINI   YES
092C 80D6                LDW   TIX   NO. INCREMENT TABLE INDEX
092D A2D2                ADD   =8
092E 70D6                STW   TIX
092F 90D6        PRCBAND LDX   TIX   TABLE INDEX
0930 80C8                LDW   BCNT
0931 A2CF                ADD   =1
0932 70C8                STW   BCNT
0933 80D4                LDW   ART3
0934 0040                SLM
0935 B837                SUB * GBT
0936 0911                SLA   1
0937 70CB                STW   ART2
0938 8838        CVORAC  LDW * GBT+1   CONSTANT VELOCITY BAND?
0939 0820                SAM
093A 1180                JMP   CONV   YES
093B E2D3                AND   =X'7FFF'   NO. MASK OUT ACC FLAG
093C 70D2                STW   OLDV
093D 0A08                SRL   8
093E 0086                SMB   VMRG   ACC BAND REACHED DESIRED
093F F005                CMW   VMRG   VELOCITY AT MERGE POINT?
0940 0880                SGR
0941 1149                JMP   ACLF   NO
0942 0086                SMB   VMRG
0943 8005                LDW   VMRG
0944 7833                STW * GBT+1   YES. SET VEL TO VMRG
0945 0100                CLR           CLEAR FRACTIONAL PARTS OF
0946 783A                STW * GBT+3   LEADING AND TRAILING EDGE
0947 783C                STW * GBT+5
0948 1180                JMP   CONV
```

```
                195 '  GREEN BAND UPDATE SUBPROGRAM  #6
  0949 0086     196 ACLF      SMB   MODE
  094A 8002     197           LDW   MODE
  094B F2D4     198           CMW   =3
  094C 0870     199           SNE
  094D 1163     200           JMP   ACCIS3
  094E 90D6     201           LDX   TIX
  094F 0040     202           SLM
  0950 8839     203           LDW * GBT+2
  0951 F2D5     204           CMW   =492
  0952 0840     205           SLS
  0953 1163     206           JMP   ACCIS3
  0954 F2D6     207           CMW   =420
  0955 0840     208           SLS
  0956 1160     209           JMP   ACCIS4
  0957 F2D7     210           CMW   =348
  0958 0840     211           SLS
  0959 115D     212           JMP   ACCIS5
  095A 82D8     213 ACCIS6    LDW   =6
  095B 70C3     214           STW   ACCTIME
  095C 1165     215           JMP   OLDACLF
  095D 82D9     216 ACCIS5    LDW   =5
  095E 70C3     217           STW   ACCTIME
  095F 1165     218           JMP   OLDACLF
  0960 82DA     219 ACCIS4    LDW   =4
  0961 70C3     220           STW   ACCTIME
  0962 1165     221           JMP   OLDACLF
  0963 82D4     222 ACCIS3    LDW   =3
  0964 70C3     223           STW   ACCTIME
  0965 70C5     224 OLDACLF   STW   AT
  0966 80D2     225           LDW   OLDV
  0967 2230     226           JSX   NEWV   CALCULATE NEW VELOCITY
  0968 0040     227           SLM
  0969 C2DB     228           ORI   =X'8000'   RESTORE ACC FLAG
  096A 90D6     229           LDX   TIX
  096B 7838     230           STW * GBT+1
  096C 80C3     231           LDW   ACCTIME
  096D 0A01     232           SRL   1
  096E 70C5     233           STW   AT
  096F 80D2     234           LDW   OLDV
  0970 2230     235           JSX   NEWV
  0971 70D8     236           STW   VAT
  0972 E2DC     237           AND   =X'FF'
  0973 70CD     238           STW   FRAC
  0974 80D8     239           LDW   VAT
  0975 0A08     240           SRL   8
  0976 0081     241           SPMUL
S 0977 2490
  0978 03C1     242           D     P100
  0979 0000     243           D     0
  097A 8179     244           LDW   $-1
  097B 70CE     245           STW   INTG
  097C A0CD     246           ADD   FRAC
  097D 70CC     247           STW   DX    DX = V(0)T + 1/2AT**2
  097E 82DD     248           LDW   =9999
  097F 1182     249           JMP   UPDATE
                250
```

```
                 '  GREEN BAND UPDATE SUBPROGRAM  #7
0980 70CC   CONV       STW   DX
0981 82DE              LDW   =99   STORE MAXIMUM FRACTION (.99)
0982 70D0   UPDATE     STW   MAXF
0983 80D6              LDW   TIX
0984 70D7              STW   TIXE
0985 2243              JSX   XLEXTE   UPDATE XLE (XLE=XLE+DX)
0986 80D6              LDW   TIX
0987 A2DF              ADD   =2
0988 70D7              STW   TIXE
0989 2243              JSX   XLEXTE   UPDATE XTE (XTE = XTE+DX)
098A 90D6              LDX   TIX   TABLE INDEX
098B 0040              SLM
098C 883B              LDW * GBT+4 CHECK OUT LEADING EDGE
098D F2E0              CMW   =608
098E 0890              SLE
098F 11C7              JMP   CHEK  TRAILING EDGE OFF DISPLAY
0990 0810              SAP
0991 0100              CLR         TRAILING EDGE WAS NEGATIVE
0992 70DA              STW   XT
0993 783B              STW * GBT+4
0994 8839              LDW * GBT+2 CHECK OUT LEADING EDGE
0995 F2E0              CMW   =608
0996 0890              SLE
0997 82E0              LDW   =608  LEADING EDGE WAS OFF DISPLAY
0998 0810              SAP
0999 1128              JMP   BEGI  LEADING EDGE NEGATIVE
099A 70D9              STW   XL
099B 7839              STW * GBT+2
099C 0A10              NOP
099D F0DA              CMW   XT
099E 0880              SGR
099F 1128              JMP   BEGI  LEADING .LE. TRAILING EDGE
09A0 80D9              LDW   XL   NEW LEADING EDGE IN FEET
09A1 F2D4              CMW   =3   GREATER THAN 3 FEET?
09A2 0880              SGR
09A3 1128              JMP   BEGI   NO. DO NOT REPRESENT THIS BAND
```

```
                 ' GREEN BAND UPDATE SUBPROGRAM #8
09A4 0100                 CLR        YES. CLEAR FLAG TO INDICATE DD OUB
09A5 70CA                 STW   DLFSBFLG   BUFFER (DBUF) IS BEING PRB
09A6 21FC                 JSX   FEETOBIT   CONVERT LEADING AND TRAILB
09A7 0040                 SLM        BAND FROM FEET TO BITS
09A8 80DE                 LDW   XTW   TRAILING WORD NUMBER (TWN)
09A9 A1B3                 ADD   STWDBUF
09AA 71B1                 STW   FILLDBUF   (FILLDBUF) = STW * DBUF +B
09AB 80DD                 LDW   XLW   LEADING WORD NUMBER
09AC B0DE                 SUB   XTW   TRAILING WORD NUMBER
09AD A2CF                 ADD   =1
09AE 70D1                 STW   NWRDFILL   NUMBER OF WORDS TO BE TURB
09AF 92E1                 LDX   =0   CLEAR INDEX TO REFERENCE PROPERB
09B0 82D1                 LDW   =-1   DD OUTPUT BUFFER (DBUF)
09B1 0000        FILLDBUF D     0   FILL WORD WITH FFFF
09B2 0401                 IXS   1   INCREMENT INDEX TO REFERENCE NEXB
09B3 7829        STWDBUF  STW * DBUF   AND SKIP (WILL ALWAYS SKIP)
09B4 0140                 CXA
09B5 F0D1                 CMW   NWRDFILL   TURNED ON ALL NECESSARY WB
09B6 0860                 SEQ
09B7 11B0                 JMP   FILLDBUF-1   NO. TURN ON NEXT WORD
09B8 90DB                 LDX   XLB   YES. LEADING BIT NUMBER
09B9 8830                 LDW * TMSK   REFERENCE APPROPRIATE MASK
09BA 0120                 INV        INVERT TO GET PROPER BAND REPRESEB
09BB A005                 ADD   TEM
09BC 90DD                 LDX   XLW   LEADING WORD NUMBER
09BD E829                 AND * DBUF   FILL LEADING WORD (EDGE) OF BB
09BE 7829                 STW * DBUF   REPRESENTATION
09BF 80DC                 LDW   XTB   TRAILING BIT NUMBER
09C0 B2CF                 SUB   =1
09C1 0130                 CAX
09C2 8830                 LDW * TMSK   REFERNCE APPROPRIATE MASK
09C3 90DE                 LDX   XTW   TRAILING WORD NUMBER
09C4 E829                 AND * DBUF   FILL TRAILING WORD (EDGE) OF B
09C5 7829                 STW * DBUF   REPRESENTATION
09C6 1128                 JMP   BEGI   NO. DO NOT REPRESENT THIS BAND
```

```
                  326 ' GREEN BAND UPDATE SUBPROGRAM  #9
   09C7 8036      327 CHEK    LDW   NBND
   09C8 F2CF      328         CMW   =1
   09C9 0870      329         SNE
   09CA 11DE      330         JMP   REDU   YES
   09CB 70C9      331         STW   COMF   NO. SET COMPRESS TABLE FLAG
   09CC 1128      332         JMP   BEGI   PROCESS NEXT BAND ENTRY
   09CD 80C9      333 FINI    LDW   COMF   YES
   09CE F2E1      334         CMW   =0   COMPRESS TABLE ?
   09CF 0860      335         SGR
   09D0 11E1      336         JMP   ASCU   NO
   09D1 80C7      337         LDW   ARTI
   09D2 A2D2      338         ADD   =8
   09D3 70C7      339         STW   ARTI
   09D4 92D2      340         LDX   =8   YES. REFERENCE SECOND BAND
   09D5 0040      341         SLM
   09D6 8837      342 TRAS    LDW * GBT
   09D7 782F      343         STW * GBT-8
   09D8 0401      344         IXS   1   INCREMENT INDEX. WILL ALWAYS SKIP
   09D9 0000      345         D     0   DUMMY INSTRUCTION
   09DA 0140      346         CXA
   09DB F0C7      347         CMW   ARTI   ALL 8-WORD BANDS MOVED UP?
   09DC 0860      348         SEQ
   09DD 11D6      349         JMP   TRAS   NO
   09DE 8036      350 REDU    LDW   NBND   REDUCE NUMBER OF
   09DF B2CF      351         SUB   =1     BANDS IN
   09E0 7036      352         STW   NBND   TABLE BY 1
        09E1      353 ASCU    EQU   $
   09E1 0039      354         BNDTRM
S  09E2 24B9
   09E3 229E      355         JSX   CMPRS  PUT BANDS IN SBUF
                  356         DO    1,11   OUTPUT 11 WORDS TO SCU
   09E4 8029      357         D311  ?
   09E5 0301
   09E6 802A
   09E7 0302
   09E8 802B
   09E9 0303
   09EA 802C
   09EB 0304
   09EC 802D
   09ED 0305
   09EE 802E
   09EF 0306
   09F0 802F
   09F1 0307
   09F2 8030
   09F3 0308
   09F4 8031
   09F5 0309
   09F6 8032
   09F7 030A
   09F8 8033
   09F9 030B
   09FA 12CD      358         JMP   GBDEXT   RETURN TO EXECUTIVE
                  359
```

```
       GREEN BAND UPDATE SUBPROGRAM  #10                        PAGE  11

                      '  GREEN BAND UPDATE SUBPROGRAM  #10
09FB 0000
09FC 61FB        FEETOBIT SUBR
09FD 8213                 LDW   SXLW
09FE 7225                 STW   STW1
09FF 722A                 STW   STW3
0A00 8214                 LDW   SXLB
0A01 7227                 STW   STW2
0A02 722C                 STW   STW4
0A03 80D9                 LDW   XL    LEADING EDGE IN FEET
0A04 2218                 JSX   WORDBIT
0A05 0040                 SLM
0A06 90DD                 LDX   XLW   LEADING WORD NUMBER
0A07 8829                 LDW * DBUF   DRIVER DISPLAY BUFFER
0A08 70D5                 STW   TEN
0A09 8215                 LDW   SXTW
0A0A 7225                 STW   STW1
0A0B 722A                 STW   STW3
0A0C 8216                 LDW   SXTB
0A0D 7227                 STW   STW2
0A0E 722C                 STW   STW4
0A0F 80DA                 LDW   XT
0A10 2218                 JSX   WORDBIT
0A11 91FB                 EXIT  FEETOBIT
0A12 2800
0A13 70DD        SXLW     STW   XLW
0A14 70DB        SXLB     STW   XLB
0A15 70DE        SXTW     STW   XTW
0A16 70DC        SXTB     STW   XTB

       GREEN BAND UPDATE SUBPROGRAM  #11                        PAGE  12

                     GREEN BAND UPDATE SUBPROGRAM  #11
0A17 0000
0A18 8217        WORDBIT  SUBR
0A19 92E1                 LDX   =0   CLEAR INDEX FOR CIRCULAR SHIFTS
0A1A 0A02                 SRL   2
0A1B 0A64        PRCDBUF  SRC 0 4   DIVIDE BY 16
0A1C 70C4                 STW   ACRT   (ACRT) = LWN OR TWN
0A1D 0140                 CXA
0A1E 0A0C                 SRL   12
0A1F 70CF                 STW   IXRT   (IXRT) = LBN OR TBN
0A20 B2CF                 SUB   =1
0A21 0820                 SAM      REMAINDER OF DIVIDE BY 16 = 0?
0A22 1229                 JMP   NOTZERO   NO
0A23 80C4                 LDW   ACRT   YES
0A24 B2CF                 SUB   =1
0A25 0000        STW1     D     0   STORE LWN OR TWN
0A26 82E2                 LDW   =16
0A27 0000        STW2     D     0   STORE LBN OR TBN
0A28 122D                 JMP   RTWRDBIT   RETURN
0A29 80C4        NOTZERO  LDW   ACRT
0A2A 0000        STW3     D     0   STORE LWN OR TWN
0A2B 80CF                 LDW   IXRT
0A2C 0000        STW4     D     0   STORE LBN OR TBN
0A2D 0217        RTWRDBIT EXIT  WORDBIT  RETURN
0A2E 2800
```

```
                412 '  GREEN BAND UPDATE SUBPROGRAM  #12
0A2F 0000
0A30 622F       413 NEWV       SUBR
0A31 E2E3       414            AND    =X'FF00'   MASK OUT FRACTIONAL PART
0A32 70CE       415            STW    INTG   STORE INTEGRAC PART
0A33 80D2       416            LDW    OLDV   INITIAL VELOCITY
0A34 E2DC       417            AND    =X'FF'   MASK OUT INTEGRAL PART
0A35 A0C5       418            ADD    AT
0A36 70CD       419            STW    FRAC   UPDATED FRACTION
0A37 F2DE       420            CMW    =99   UPDATED FRAC GREATER THAN 99?
0A38 0380       421            SGR
0A39 123F       422            JMP    L99   NO
0A3A 80CE       423            LDW    INTG   YES. ADD 1 FOOT/SEC
0A3B A2E4       424            ADD    =X'100'   TO INTEGRAL
0A3C 70CE       425            STW    INTG   PART
0A3D 80CD       426            LDW    FRAC   SUBTRACT 100 FROM TOTAL FRAC
0A3E B2DE       427            SUB    =99   TO GET REMAINDER
0A3F A0CE       428 L99        ADD    INTG   ADD INTEGRAL TO FRAC PART
0A40 922F       429            EXIT   NEWV
0A41 2800
                430 *
0A42 0000
0A43 6242       431 XLEXTE     SUBR
0A44 0040       432            SLM
0A45 00D7       433            LDX    TIXE
0A46 883A       434            LDW *  GBT+3   FRACTIONAL PART OF XLE OR XTE
0A47 A0CC       435            ADD    DX   CHANGE IN BAND POSITION
0A48 70CD       436            STW    FRAC   UPDATED FRACTION
0A49 F0D0       437            CMW    MAXF   UPDATED FRAC.GT.99 OR 9999?
0A4A 0380       438            SGR
0A4B 1252       439            JMP    L9S
0A4C 8839       440            LDW *  GBT+2
0A4D A2CF       441            ADD    =1
0A4E 7839       442            STW *  GBT+2
0A4F 80CD       443            LDW    FRAC   SUB 100 OR 10000 FROM UPDATED
0A50 B0D0       444            SUB    MAXF   FRACTION TO GET REMAINDER
0A51 A2CF       445            ADD    =1
0A52 783A       446 L9S        STW *  GBT+3
0A53 9242       447            EXIT   XLEXTE   RETURN
0A54 2800
                448
```

```
               ' MERGE AREA DETECTOR ROUTINE     #13
               *
0A55 0000      PDCLEAR  D      0  FLAG SET TO 1 IF ANY PD ON
0A56 0000
0A57 6256      CKPD     SUBR
0A58 0040               SLM
0A59 0100               CLR
0A5A 7255               STW    PDCLEAR
0A5B 92E5               LDX    =-8
0A5C 0080      HERE     SMB    SPPS
0A5D 3978               LDW *  SPPS+8   PRESENCE DETECTOR ACTIVATED?
0A5E 0800               SAZ
0A5F 1263               JMP    PACT   YES
0A60 0401      INCR     IXS    1    NO. ALL DETECTORS CHECKED?
0A61 125C               JMP    HERE  NO. UP IXR AND CHECK NEXT P.D.
0A62 1276               JMP    RTPD
0A63 92CF      PACT     LDW    =1
0A64 7255               STW    PDCLEAR
0A65 0080               SMB    TPPS
0A66 3980               LDW *  TPPS+8   COUNTS SINCE TURN ON
0A67 0820               SAM       ALL NEGATIVE TPPS ARE TOO
0A68 126A               JMP    PCT1   LONG
0A69 1274               JMP    SBKFG
0A6A 0080      PCT1     SMB    SPPS
0A6B 3978               LDW *  SPPS+8
0A6C 0A01               SRL    1
0A6D 0800               SAZ       ALL SPPS GREATER THAN 1
0A6E 1274               JMP    SBKFG   ALSO TOO LONG
0A6F 0080               SMB    TPPS
0A70 3980               LDW *  TPPS+8
0A71 F0C2               CMW    PDMAX
0A72 0880               SGR
0A73 1260               JMP    INCR
0A74 0100      SBKFG    CLR       MERGE AREA OCCUPIED
0A75 127E               JMP    SFG     MASK BANDS OFF
0A76 0086      RTPD     SMB    K51F
0A77 800C               LDW    K51F
0A78 0800               SAZ
0A79 127D               JMP    ALLOK
0A7A 8255               LDW    PDCLEAR
0A7B 0800               SAZ
0A7C 1280               JMP    ECKPD
0A7D 82D1      ALLOK    LDW    =-1    MERGE AREA CLEAR
     0A7E      SFG      EQU    $
0A7E 0086               SMB    K51F
0A7F 700C               STW    K51F
     0A80      ECKPD    EQU    $   MASK FLAG SET FOR PD
0A80 02F5               DIN    15,5   NOW CHECK FOR RAMP ACTIVITY
0A81 0830               SAO         R11 SENSOR USED
0A82 128D               JMP    R11OFF
0A83 0100               CLR         R11 ON
0A84 7290               STW    SCR11   RESET SEC COUNT
0A85 0080               SMB    TIME   AND CLOCK FOR ELAPSED SEC
0A86 8021               LDW    TIME
0A87 728F               STW    R11C
0A88 82D1               LDW    =-1
0A89 0086               SMB    PSF   UNMASK POWER SAVE
```

```
0A8A 700D    505             STW    PSF
     0A8B    506 CKPDXT      EQU    $
0A8B 9256    507             EXIT   CKPD
0A8C 2800
     0A8D    508 R11OFF      EQU    $
0A8D 008F    509             CUSC
0A8E 26E8
0A8F 0000    510 R11C        D      0  CLOCK LAST SEC ELAPSED
0A90 0000    511 SCR11       D      0    SEC COUNT
0A91 001E    512 BACT        D      30 SEC LAND DISPLAY AFTER R11 ON
0A92 1294    513             JMP    $+2    OVERFLOW
0A93 128B    514             JMP    CKPDXT  NO SEC COUNT TIMEOUT
0A94 0100    515             CLR      MASK BANDS
0A95 0086    516             SMB    PSF
0A96 700D    517             STW    PSF   VIA POWER SAVE MASK
0A97 8291    518             LDW    BACT  KEEP CUSC OVERFLOWING
0A98 7290    519             STW    SCR11     UNTIL RESET BY
0A99 828F    520             LDW    R11C      R11 ACTIVATION
0A9A B2E6    521             SUB    =500
0A9B 728F    522             STW    R11C
0A9C 128B    523             JMP    CKPDXT
             524
```

```
                ' COMPRESS DBUF TO MAKE SBUF  #14
                *
                * COMPRESS DBUF(10 WORDS) TO SBUF(5 WORDS)
                *  TAKE OUT EVERY OTHER BIT
                *
0A9D 0000
0A9E 629D         CMPRS     SUBR
0A9F 0040                   SLM
0AA0 0100                   CLR      CLEAR LEFT HALF FLAG
0AA1 72C9                   STW     CMPRSFL
0AA2 92E7                   LDX     =-5  NUMBER OF WORDS IN SBUF BAND
     0AA3         RPT       EQU     $
0AA3 62CA                   STX     ISV1
0AA4 0A31                   SLL D 1        CONVERT TO INDEX FOR DBUF
0AA5 8834                   LDW * DBUF+11  SECOND OF TWO DBUF WORDS
0AA6 72CB                   STW     BRAVO    SAVE IT MOMENTARILY
0AA7 8833                   LDW * DBUF+10  FIRST OF TWO, CRUNCH IT
     0AA8         AGIN      EQU     $
0AA8 92E1                   LDX     =0        SAVE ONLY ALTERNATE BITS
0AA9 0A22                   SRL D 2        SHIFT BITS TO LEFT HALF WORD
0AAA 0A01                   SRL     1
0AAB 0A21                   SRL D 1
0AAC 0A01                   SRL     1
0AAD 0A21                   SRL D 1
0AAE 0A01                   SRL     1
0AAF 0A21                   SRL D 1
0AB0 0A01                   SRL     1
0AB1 0A21                   SRL D 1
0AB2 0A01                   SRL     1
0AB3 0A21                   SRL D 1
0AB4 0A01                   SRL     1
0AB5 0A21                   SRL D 1
0AB6 0A01                   SRL     1
0AB7 0A21                   SRL D 1
0AB8 82C9                   LDW     CMPRSFL  CHECK FLAG IF THIS FIRST
0AB9 D2CF                   ORE     =1
0ABA 72C9                   STW     CMPRSFL
0ABB 0800                   SAZ              SKIP IF SECOND DBUF WORD
0ABC 12C6                   JMP     PT2      GO DO SECOND,SAVE FIRST
0ABD 0A38                   SLL D 8          MOVE SECOND INTO ACC
0ABE C2CC                   ORI     LFTHLF   PUT IN THE OTHER HALF
0ABF 92CA                   LDX     ISV1     PUT IT IN SBUF
0AC0 0084                   SMB     SBUF
0AC1 73CD                   STW * SBUF+5
0AC2 0401                   IXS     1        COUNT IF ANOTHER TO DO
0AC3 12A3                   JMP     RPT       YES THERE IS MORE
0AC4 929D                   EXIT    CMPRS       NO MORE TO DO, BYE-BYE
0AC5 2800
     0AC6         PT2       EQU     $
0AC6 62CC                   STX     LFTHLF   TUCK FIRST CRUNCHED AWAY
0AC7 82CB                   LDW     BRAVO    GET THE SECOND WORD
0AC8 12A8                   JMP     AGIN     GO CRUNCH IT
0AC9              CMPRSFL   RES     1
0ACA              ISV1      RES     1
0ACB              BRAVO     RES     1
0ACC              LFTHLF    RES     1
```

```
               '  GREEN BAND UPDATE SUBPROGRAM  #15
               *
0ACD 0080      GBDEXT   SMB   EXEK
0ACE 1025               JMP   EXEK   RETURN TO EXECUTIVE
               *
                        END

0ACF 0001
0AD0 FFF6
0AD1 FFFF
0AD2 0008
0AD3 7FFF
0AD4 0003
0AD5 01EC
0AD6 01A4
0AD7 015C
0AD8 0006
0AD9 0005
0ADA 0004
0ADB 8000
0ADC 00FF
0ADD 270F
0ADE 0063
0ADF 0002
0AE0 0260
0AE1 0000
0AE2 0010
0AE3 FF00
0AE4 0100
0AE5 FFF8
0AE6 01F4
0AE7 FFFB
```

NO ERRORS

ACCIS3	0963	ACCIS4	0960	ACCIS5	095D	ACCIS6	095A
ACCTIME	08C3	ACLF	0949	ACRT	08C4	AGIN	0AA8
ALLG	08AF	ALLOK	0A7D	ART1	08C7	ART2	08CB
ART3	08D4	ASCU	09E1	AT	08C5	BACT	0A91
BCNT	08C8	BEGI	0928	BKFL	08C6	ENDTRM	24B9
BRAVO	0ACB	CHEK	09C7	CKPD	0A57	CKPDXT	0A8B
CMPRS	0A9E	CMPRSFL	0AC9	COMF	08C9	CONV	0980
CUSC	3EE8	CVORAC	0938	DBFSBFLG	08CA	DEUF	0829
DX	08CC	ECKPD	0A80	EXEK	0025	FEETOBIT	09FC
FILLDBUF	09B1	FINI	09CD	FRAC	08CD	GBDEXT	0ACD
GBT	0837	GBUD	08F8	HERE	0A5C	INCR	0A60
INFG	1804	INIT	08E0	INT	0020	INTG	08CE
ISVI	0ACA	IXRT	08CF	K51F	180C	L9S	0A3F
L9S	0A52	LAMJ	0800	LFTHLF	0ACC	LST	24EA
MAXF	08D0	MODE	1802	NBND	0836	NEWV	0A30
NOTZERO	0A29	NWRDFILL	08D1	OLDACLF	0965	OLDV	08D2
P100	08C1	PACT	0A63	PCTI	0A6A	PDCLEAR	0A55
PDMAX	08C2	PRCBAND	092F	PRCDBUF	0A1B	PSF	180D
PT2	0AC6	R11C	0A8F	R11OFF	0A8D	REDU	09DE
RPT	0AA3	RTPD	0A76	RTWRDBIT	0A2D	SAVE	08D3
SBKFG	0A74	SBUF	10C8	SCR11	0A90	SETC	0918
SFG	0A7E	SPMUL	0490	SPPS	0170	STW1	0A25
STW2	0A27	STW3	0A2A	STW4	0A2C	STWDBUF	09B3
SXLB	0A14	SXLW	0A13	SXTB	0A16	SXTW	0A15
TEM	08D5	TIME	0021	TIX	08D6	TIXE	08D7
TMSK	08B0	TPPS	0178	TRAS	09D6	UPDATE	0982
VAT	08D8	VMRG	1805	WORDBIT	0A18	WORKFLG	247A
XL	08D9	XLB	08DB	XLEXTE	0A43	XLW	08DD
XT	08DA	XTB	08DC	XTW	08DE		

PAS?

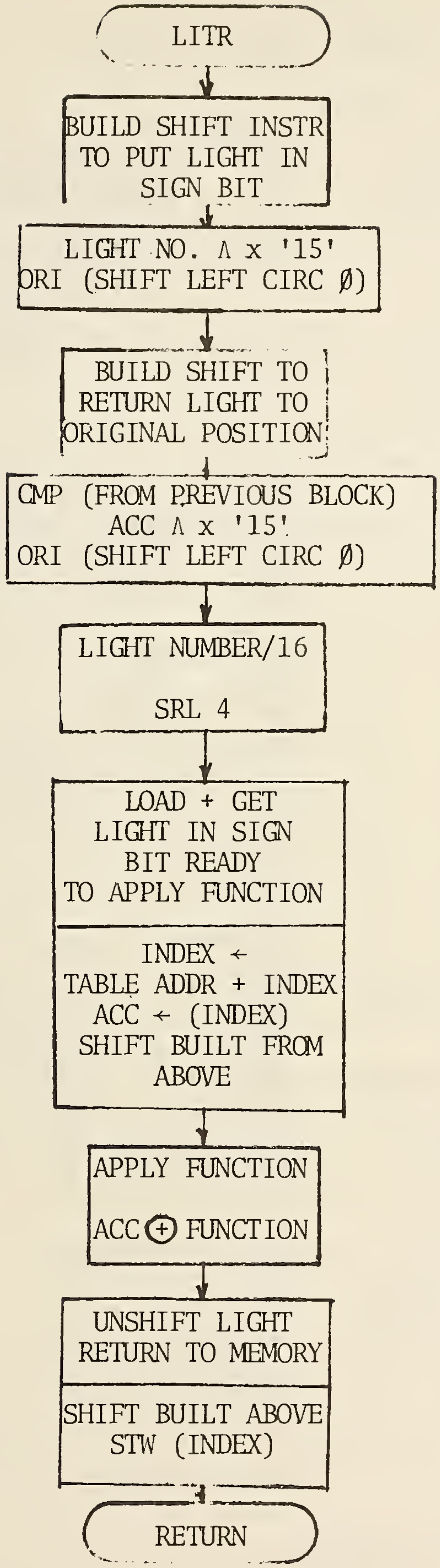
CHECK LIGHT IN BAND SUBROUTINE

Routine LITR was added for use by the band-trim subroutine. It will retrieve a selected light from a display and put it in the sign list of the accumulator. Depending upon the function selected, light status or change light state, it will return the result in the accumulator.

A standard SUBR entry is used with 3 calling parameters:

1. Address of a 10 - word display ,
2. Desired light in display (0-159), and
3. Function in sign list
   - \- Flip light state
   - \+ Return light state in accumulator sign list
     - \- Light on
     - \+ Light off.

The routine builds shift and unshift instructions from the light number so the display word will be ordered correctly when returned to the data area. Then the index to the word containing the selected light is computed by dividing the light number by 16 (shift right 4) and the word retrieved, shifted, function list exclusive - or'ed with the light, unshifted and returned to the display. The product of the exclusive - or operation is reloaded and control returned to call.


LITR
BUILD SHIFT INSTR TO PUT LIGHT IN SIGN BIT
LIGHT NO. Λ x '15'
ORI (SHIFT LEFT CIRC Ø)
BUILD SHIFT TO RETURN LIGHT TO ORIGINAL POSITION
CMP (FROM PREVIOUS BLOCK)
ACC Λ x '15'
ORI (SHIFT LEFT CIRC Ø)
LIGHT NUMBER/16
SRL 4
LOAD + GET LIGHT IN SIGN BIT READY TO APPLY FUNCTION
INDEX ←
TABLE ADDR + INDEX
ACC ← (INDEX)
SHIFT BUILT FROM ABOVE
APPLY FUNCTION
ACC ⊕ FUNCTION
UNSHIFT LIGHT RETURN TO MEMORY
SHIFT BUILT ABOVE
STW (INDEX)
RETURN

```
                'CHECK LIGHT IN BAND
                *
                *  CALL:
                *          LITR
                *(0)       D      A(LIGHT TABLE, 10 WORDS)
                *(1)       D      LIGHT NUMBER, 0-159
                *(2)       D      FUNCTION
                *                 -  FLIP LIGHT STATE
                *                 +  RETURN LIGHT STATE IN ACCUMULATOR
                *                    -  LIGHT ON
                *                    +  LIGHT OFF
                *(3)       RETURN
                *
15E2 0000
15E3 65E2       LITR       SUBR
15E4 8801                  LDW * 1     LIGHT NUMBER
15E5 E5FF                  AND    FFTN  BIT POSITION WITHIN WORD
15E6 C5FE                  ORI    SLC0  BUILD SHIFT TO MOVE IT TO SIGN
15E7 75F4                  STW    SHFTO
15E8 0110                  CMP          BUILD SHIFT TO UNSHIFT WORD
15E9 E5FF                  AND    FFTN
15EA C5FE                  ORI    SLC0
15EB 75F7                  STW    SHFTU
15EC 8802                  LDW * 2     GET FUNCTION
15ED E5FD                  AND    GN    CLEAN TO LEAVE ONLY SIGN
15EE 75FC                  STW    A    SAVE IT
15EF 8801                  LDW * 1   WORD POSITION IN TABLE
15F0 0A04                  SRL    4
15F1 A800                  ADD * 0    + TABLE ADDRESS
15F2 0130                  CAX
15F3 8800                  LDW * 0   GET WORD WITH LIGHT BIT
15F4 0A10       SHFTO      NOP        SHIFT BIT TO SIGN
15F5 D5FC                  ORE    A   LAY FUNCTION ON IT
15F6 75FC                  STW    A   SAVE IT FOR RETURN
15F7 0A10       SHFTU      NOP        UNSHIFT IT BACK
15F8 7800                  STW * 0  AND PUT IT AWAY
15F9 85FC                  LDW    A   GET LIGHT BACK
15FA 95E2       XT         EXIT   LITR,3
15FB 2803
15FC            A          RES    1
15FD 8000       GN         D      X'8000'
15FE 0A50       SLC0       SLC    0
15FF 000F       FFTN       D      15
                           ORIG   X'1600'
                           END
```



```
A          15FC  FFTN       15FF  GN         15FD  LITR       15E3
SHFTO      15F4  SHFTU      15F7  SLC0       15FE  XT         15FA
PAS?
```

GREEN BAND STATUS SUBPROGRAM

The power-save flag (PSF) was added to the global data and program changes were:

1. Ramp clear check, either R7 sensor or R6B sensor having no activation in 10 sec;
2. Back-up of sensor R6B by R7, if R7 has been activated 15 seconds ago and no R6A activation has been detected, an R6B pseudo-activation is created;
3. During SG to MM transition the ramp clear check is used to inhibit the transition until the clear is given;
4. Vestigial data and program segments were removed;
5. R4 sensor was added to R3 check for yield sign: and
6. Congestion check using R5-R1, 4 second occupancy, and M1-M2, 10 second occupancy, for either condition the maximum wait-in-red time for the traffic signal is set to 20 seconds.

Local data additions at the beginning are R7TOG, the indicator for change 2; NAXREDP, the 20 second constant for change 6; and RCLR, the ramp clear indicator for change 1.

Added data initializations were R7TOG, RCLR, MCLK, and MCNT (clock and second count for the congestion check in the merge area), RCLK and RCNT (for the ramp), VWCLK and VWCNT (for the R7 back-up of R6B), SCLK nad SCNT (for the ramp clear check), MAS (merge area sensor check), and RMPS (ramp sensor check).

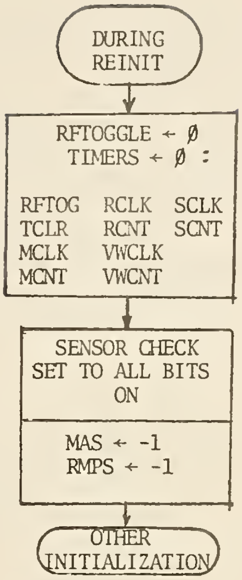
In the GBS main entry code, the ramp clear and R7 back-up code were inserted. Ramp clear is indicated by R6B or R7 having been clear SLIM seconds. RCLR is set to zero for clear and one for not clear.

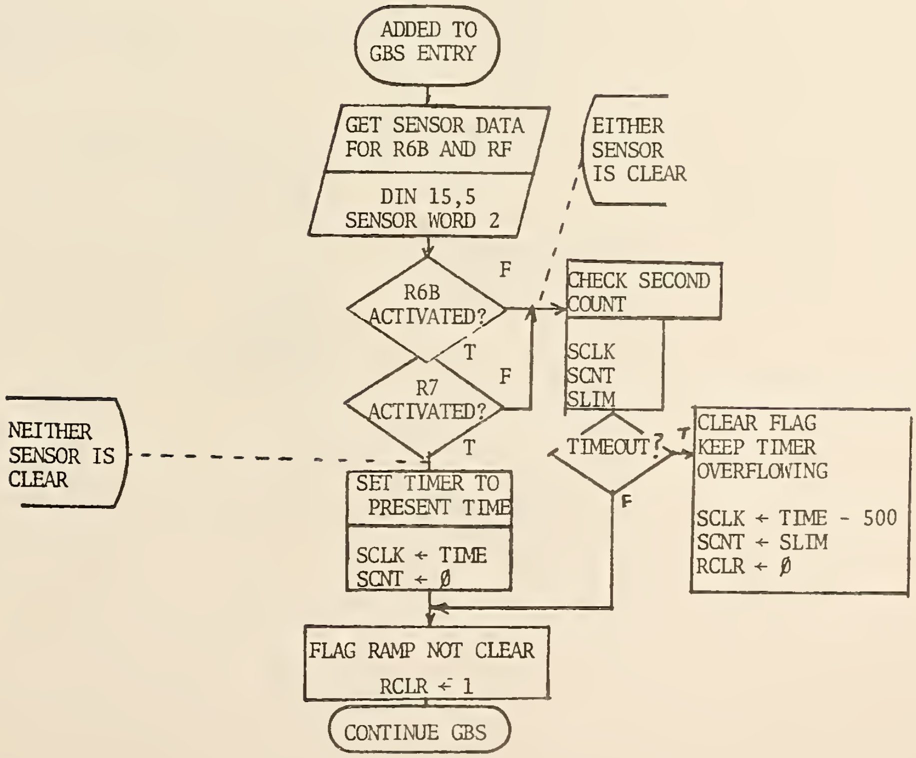
The accompanying code using the RCLR flag is the modified

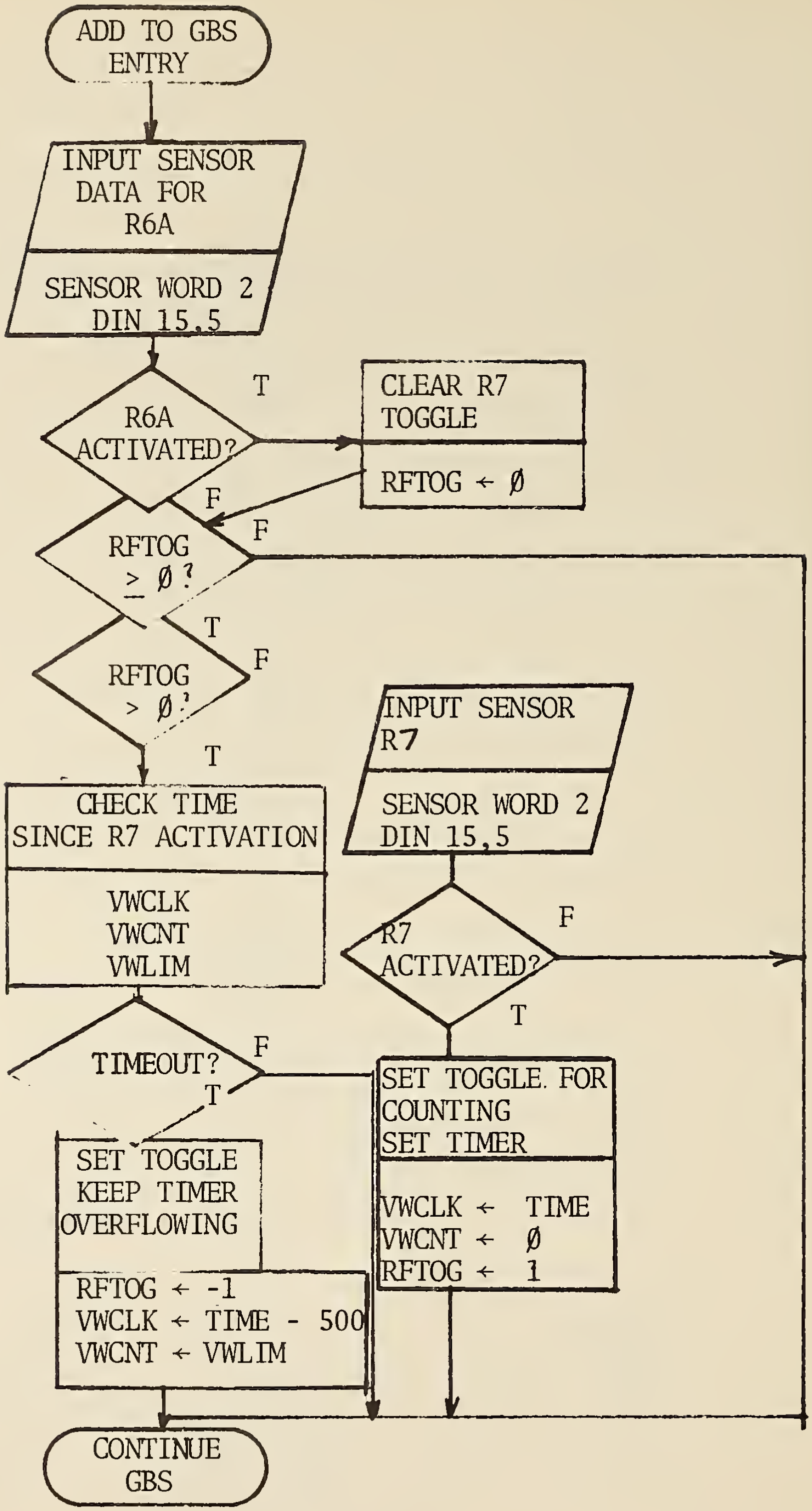
section at label SETMM. If the ramp is not clear (RCLR=1) the change is not allowed and WREDFLAG is set to -1 to keep the control from getting involved in TWRED code (preceeding SETMM). For the R6B back-up, R6A is checked because it being activated clears the check (R7TOG = zero). When R7 is activated R7TOG is set to one and the timer counts until VWLIM seconds have elapsed. If it overflows R6A has not been activated so R7TOG is set to negative one and remains there until the R6A activation clears it. Associated with the flag control is added code in CKDEMAND; R6B is checked and if there is a vehicle waiting it proceeds normally. If it takes the "no-vehicle-waiting" execution, it will now execute the R7TOG check for a time-out and take the same procedure for the vehicle-found-waiting condition if R7TOG = -1.

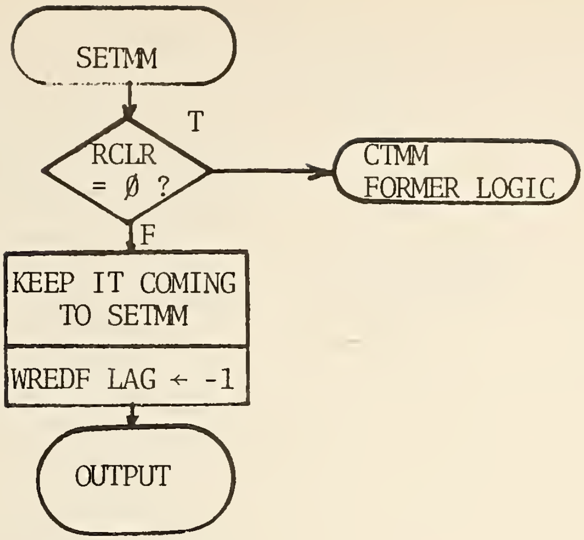
At CHKK19 following the present volume setting of the MAXRED time, a check is made for ramp congestion (R5 to R1 being occupied for RLIM seconds) or merge area congestion (M1 or M2 being occupied for MLIM seconds). The storage for the ramp sensor status is RMPS and starts execution (or is reset) with all bits set. The ramp sensor status is input, "anded" with RMPS until RMPS goes to zero indicating all the ramp sensors have been occupied and should the timer (RCNT) overflow before the flags all indicate zero, MAXRED is set to MAXREDQ value (congestion). Using the same logic M1 and M2 are handled in MERCHR through MAS (merge area sensors) and MLIM (seconds to overflow = congestion.) Whenever the sensors go clear indicating movement, the counts are reset, the RMPS and MAS storage set to -1 (all bit sets) and the checks start over.

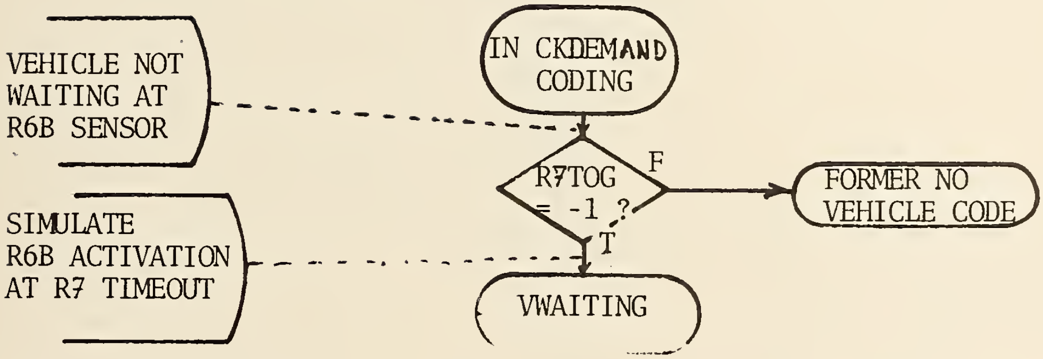
In the MWC code for the yield sign, starting at CKØ15, R4 is input and checked for a green band alongside if activated (DBUFT 6 "anded" with TR4DM). If R4 is activated and the band not on alongside, the yield sign is activated.

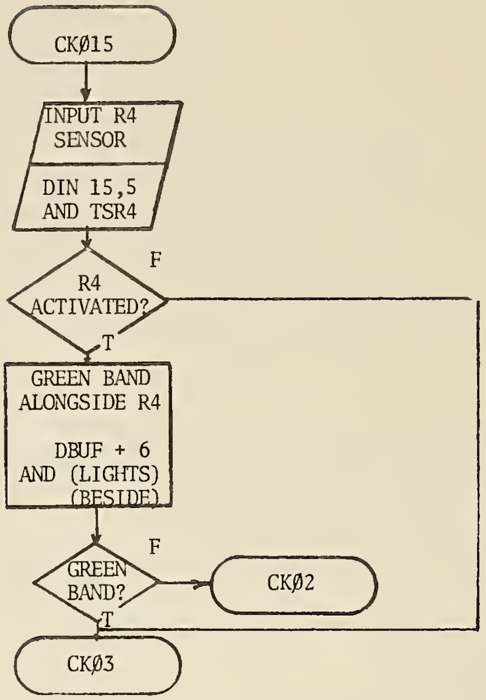

DURING REINIT
RFTOGGLE ← Ø
TIMERS ← Ø :
RFTOG RCLK SCLK
TCLR RCNT SCNT
MCLK VWCLK
MCNT VWCNT
SENSOR CHECK SET TO ALL BITS ON
MAS ← -1
RMPS ← -1
OTHER INITIALIZATION

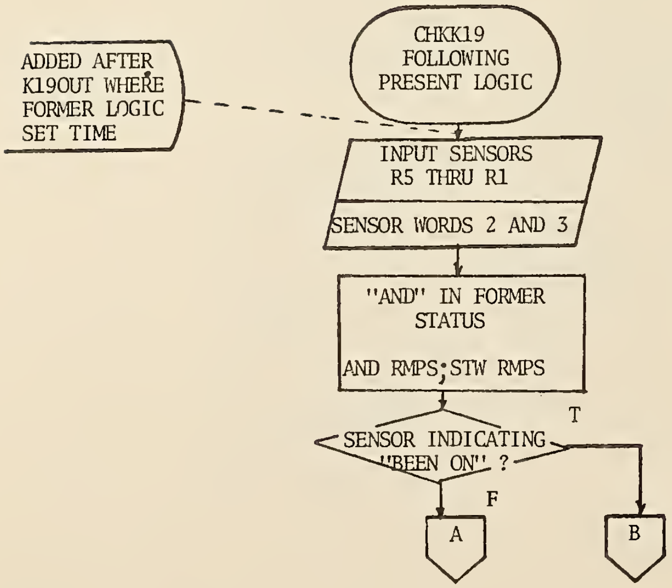

ADDED TO GBS ENTRY
GET SENSOR DATA FOR R6B AND RF
DIN 15,5
SENSOR WORD 2
EITHER SENSOR IS CLEAR
R6B ACTIVATED?
F
T
CHECK SECOND COUNT
SCLK
SCNT
SLIM
R7 ACTIVATED?
F
T
NEITHER SENSOR IS CLEAR
TIMEOUT?
T
F
CLEAR FLAG KEEP TIMER OVERFLOWING
SCLK ← TIME - 500
SCNT ← SLIM
RCLR ← Ø
SET TIMER TO PRESENT TIME
SCLK ← TIME
SCNT ← Ø
FLAG RAMP NOT CLEAR
RCLR ← 1
CONTINUE GBS

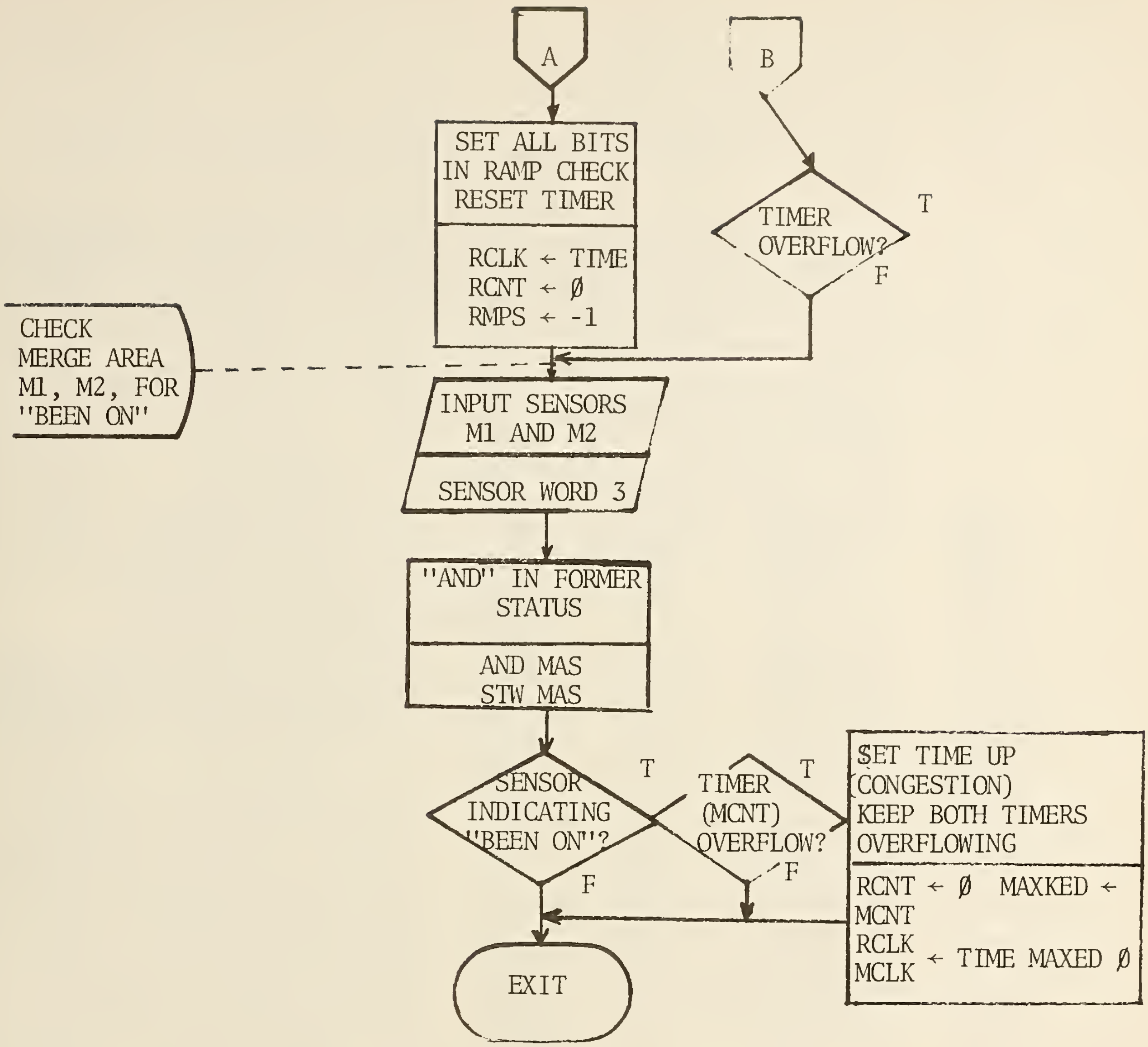

ADD TO GBS ENTRY
INPUT SENSOR DATA FOR R6A
SENSOR WORD 2 DIN 15,5
R6A ACTIVATED?
T
F
CLEAR R7 TOGGLE
RFTOG ← Ø
RFTOG ≥ Ø ?
F
T
RFTOG > Ø ?
F
T
INPUT SENSOR R7
SENSOR WORD 2 DIN 15,5
CHECK TIME SINCE R7 ACTIVATION
VWCLK
VWCNT
VWLIM
R7 ACTIVATED?
F
T
TIMEOUT?
F
T
SET TOGGLE FOR COUNTING
SET TIMER
VWCLK ← TIME
VWCNT ← Ø
RFTOG ← 1
SET TOGGLE
KEEP TIMER
OVERFLOWING
RFTOG ← -1
VWCLK ← TIME - 500
VWCNT ← VWLIM
CONTINUE GBS


SETMM
RCLR = Ø ?
T
CTMM FORMER LOGIC
F
KEEP IT COMING TO SETMM
WREDF LAG ← -1
OUTPUT


VEHICLE NOT WAITING AT R6B SENSOR
SIMULATE R6B ACTIVATION AT R7 TIMEOUT
IN CKDEMAND CODING
R7TOG = -1 ?
F
FORMER NO VEHICLE CODE
T
VWAITING


CKØ15
INPUT R4 SENSOR
DIN 15,5 AND TSR4
R4 ACTIVATED?
F
T
GREEN BAND ALONGSIDE R4
DBUF + 6 AND (LIGHTS) (BESIDE)
GREEN BAND?
F
T
CKØ2
CKØ3


ADDED AFTER K19OUT WHERE FORMER LOGIC SET TIME
CHKK19 FOLLOWING PRESENT LOGIC
INPUT SENSORS R5 THRU R1
SENSOR WORDS 2 AND 3
"AND" IN FORMER STATUS
AND RMPS; STW RMPS
SENSOR INDICATING "BEEN ON" ?
T
F
A
B


A
B
SET ALL BITS IN RAMP CHECK RESET TIMER
RCLK ← TIME
RCNT ← Ø
RMPS ← -1
TIMER OVERFLOW?
T
F
CHECK MERGE AREA M1, M2, FOR "BEEN ON"
INPUT SENSORS M1 AND M2
SENSOR WORD 3
"AND" IN FORMER STATUS
AND MAS
STW MAS
SENSOR INDICATING "BEEN ON"?
T
F
TIMER (MCNT) OVERFLOW?
T
F
SET TIME UP (CONGESTION) KEEP BOTH TIMERS OVERFLOWING
RCNT ← Ø MAXKED ← MCNT
RCLK MCLK ← TIME MAXED Ø
EXIT

```
                ' GREEN BAND STATUS PROGRAM  #1
                *     VERSION PREPARED 11-3-74/3-28-77
                *
                *     THIS ASSEMBLY OF GBS INCLUDES THE TAMPA SITE
                *     ETA'S FOR THE SG MODE BANDS, AND THE LOGIC
                *     TO PROVIDE THE BAND EDGE LOCATIONS FOR THE
                *     VARIABLE SPEED GREEN BANDS USED ON THE
                *     TAMPA RAMP IN THE MOVING MERGE MODE.
                *
                *     INCLUDES CALLS TO TABLE COPY ROUTINE
                *     THIS VERSION ALSO HAS INIT PRINT LOGIC.
                *
                          ORIG   X'1800'
     0021        TIME     EQU    X'21'      2 MSEC CLOCK
     0829        DBUF     EQU    X'829'     DD OUTPUT BUFFER
     10BF        DSTA     EQU    X'10BF'    SD OUTPUT BUFFER
     10C8        SBUF     EQU    X'10C8'    SD OUTPUT BUFFER
     2479        ALLG     EQU    X'2479'    ALL GREEN FLAG
     2000        JVELVL   EQU    X'2000'    GB VEL-VOL ROUTINE
     2001        BIXR     EQU    X'2001'    BEG POINTER TO VEL TABLE
     2002        CIXR     EQU    X'2002'    CURRENT INDEX TO VEL TABLE
     2003        TVL      EQU    X'2003'    3 MINUTE TOTAL VELOCITY
     2004        NUMV     EQU    X'2004'
     0020        JFRM     EQU    X'20'      INIT RETURN
     0023        EXEC     EQU    X'23'      EXEC RETURN ADDRESS
     2400        NBND     EQU    X'2400'   NUMBER OF BANDS IN TABLE
     0836        RLNBND   EQU    X'836'    ACTUAL LOC OF NBND IN GBUD
     2401        GBT      EQU    X'2401'   GREEN BAND TABLE
     0BC8        PNTW     EQU    X'BC8'    TABLE OF HIWY VEH LIST PNTRS
     247C        GBSSTRT  EQU    X'247C'   START TABLE COPY
     248F        GBSDONE  EQU    X'248F'   TABLE COPY AT RETURN
     0490        SPMUL    EQU    X'490'    SOFTWARE MULTIPLY
     04E0        SPDIV    EQU    X'4E0'    SOFTWARE DIVIDE
     0048        RLTQUE   EQU    X'48'     RAMP QUEUE STATUS WORD
     3EE8        CUSC     EQU    X'3EE8'   CHECK SEC COUNT
1800 10AD                 JMP    INI
1801 10F1                 JMP    GBS
                *
                * TAMPA SITE GEOMETRY
                *      RAMP DISPLAY LENGTH IS 608 FEET
                *      DISPLAY END TO NOSE IS 32 FEET
                *      NOSE TO MP IS 204 FEET
                *      STOP LINE IS 264 FEET ALONG DISPLAY
                *      SG BAND LAUNCH POINTS
                *           LEADING EDGE AT 276 FEET
                *           TRAILING EDGE AT 244 FEET
```

```
               ' GREEN BAND STATUS   #2
1802 0000      MODE      D      0   SM(1), SG(2), OR MM(3)
1803 0000      Q         D      0   3 MINUTE VOLUME
1804 0000      INFG      D      0   INITIALIZATION FLAG
1805 0000      VMRG      D      0   DESIRED VELOCITY AT MERGE POINT
1806 0000      VBAR      D      0   3 MINUTE AVERAGE HWY VEL
1807 0000      K30F      D      0   K30 ACTIVE FLAG
1808 0000      K30V      D      0   VEL OF VEH THAT ACTIVATED K30
1809 0000      ACTF      D      0   SENSOR ACTIVE FLAG
180A 0000      MINCLK    D      0   ONE   MINUTE CLOCK
180B 0000      TNMSCK    D      0   TEN MSEC CLOCK
180C FFFF      K51F      D      -1   MASK FOR K51  ACTIVATION
180D FFFF      PSF       D      -1   POWER SAVE MASK
               *
               * UP TO HERE, EXTERNAL DATA
               *
180E 0000      SMSGBD    D      0   SM SG VBAR BOUNDARY
180F 0000      SGMMBD    D      0   SG MM VBAR BOUNDARY
1810 0000      SSFLAG    D      0   3 MINUTE FLAG
1811 0000      INTTIM    D      0   INITIALI¯ATION TIME
1812 0000      ARTI      D      0   INIT TIME INTERNAL TO GBS
1813 0000      EXPFLG    D      0
1814 0000      GRNOPTFL  D      0   OPERATOR OPTION FLAG
1815 0000      LIGHT     D      0   STATE OF TRAFFIC LIGHT
1816 0000      TIMEGRN   D      0   TIME LIGHT TURNED GREEN
1817 0000      TIMEAMBR  D      0   TIME LIGHT SET AMBER
1818 0000      TIMERED   D      0   TIME LIGHT TURNED RED
1819 09C4      MAXGREEN  D      X'9C4'   5 SEC. GREEN
181A 01F4      MAXAMBR   D      X'1F4'   1 SECOND AMBER
181B 0FA0      MAXRED    D      X'0FA0'
181C 0000      MINRED    D      0
181D 03E8      SGMINRED  D      X'3E8'   SG MIN.RED
181E 0000      WREDFLAG  D      0    'WAIT IN RED' FLAG
181F 0FA0      WREDTIME  D      X'0FA0'   WAIT IN RED TIME
1820 0000      MMSGFG    D      0   MM  SG FIRST TIME THROUGH FLAG
1821 0000      MMSGAM    D      0   TIME LIGHT SET AMBER IN MM  SG
1822 0000      SGMMT     D      0
1823 0000      SGMMFG    D      0
1824 0000      INTREDFL  D      0   INIT RED FLAG (USED IN SM AND SG)
1825 0000      VWAITFLG  D      0   VEHICLE WAIT FLAG
1826 0000      VWAITIME  D      0   T VEH WAIT START AT DEMAND LOOP
1827 0000      FAON      D      0   TIME FLASH AMBER LIGHT TURNED ON
1828 0000      FAOF      D      0   TIME FLASH AMBER LIGHT TURNED OFF
1829 0000      FAFG      D      0   FLASH AMBER LIGHT ON-OFF FLAG
182A 0000      BEGP      D      0   BEGINNING POINTER
182B 0000      ENDP      D      0   END POINTER
```

```
               ' GREEN BAND STATUS    #3
182C 0000        K19T      DATA  0
182D 03E8        K19QSG    D     X'3E8'   2 SEC MAX WAIT IF QUEUE
182E 1D4C        K19NOQSG  D     X'1D4C'  15 SEC MAX WAIT IF NO QUEUE
182F 03E8        K19QSM    D     X'3E8'    2 SEC
1830 09C4        K19NOQSM  D     X'9C4'    5 SEC
1831 0004        FOUR      DATA  4
1832 07D0        SEC4      DATA  X'7D0'   4 SEC.
1833 0000        OLDS      DATA  0
1834 0000        SIXR      D     0
1835 0000        BIND      D     0   BAND INDICATOR
1836 0000        D         D     0
1837 0000        GLTA      D     0
1838 0000        GLTI      D     0
1839 0000        LETA      D     0   LEADING ETA (7/9 1.024)
183A 0000        LLEN      D     0   LEADING LENGTH IN SEC (7/9 1.024)
183B 0000        FPBC      D     0   PROCESSED BAND COUNTER
183C 0000        LTI       D     0   LEADING TI (7/9 1.024)
183D 0000        FTIX      D     0   TABLE INDEX - USED IN FILTER
183E 0000        LDFG      D     0   LAUNCH-DELETE FLAG
183F 0041        QSGMM     D     65   (1300 VEH PER HR)
1840 0000        LTA       D     0
1841 0000        TTA       D     0
1842 0000        LTAMLHDY  D     0
1843 0000        TTAPTHDY  D     0
1844 0000        LHDYTLLN  D     0    0 SEC
1845 0000        THDYTLLN  D     0    0 SEC
1846 0000        LHDYTLFL  D     0
1847 0000        THDYTLFL  D     0
1848 00FA        RGPTOLLN  D     X'FA'  RGAP TOLERANCE (LAUNCH .5 SEC)
1849 0000        RGPTOLFL  D     0   RGAP TOLERANCE (FILTER)
184A 0000        RGAPTOL   D     0
184B 007D        BINKTOL   D     X'7D'   MOVING MODE TOLERANCE
184C 0000        LVEHFLAG  D     0   FIRST VEHICLE FLAG
184D 004B        QMMSG     D     75  (1500 VEH/HR SEL. BY CITY OF TAMP
184E 034C        RMPL      D     844   TAMPA GB DISP START TO MP
184F 00C8        LHDY      D     X'C8'   LEADING HEADWAY (.4 SECS)
1850 03E8        THDY      D     X'3E8'   TRAILING HEADWAY (2 SECS)
1851 0000        LTHDY     D     0
1852 0000        LVEL      D     0   LEADING VELOCITY (9/7)
1853 0000        REDFLAG   D     0
```

```
               ' GREEN BAND STATUS    #4
1854 0000      MGAP      D      0   MEASURED GAP
1855 0000      RGAP      D      0   REQUIRED GAP
1856 0000      TMTM      D      0
1857 0000      SVEL      D      0   VELOCITY (9/7)
1858 0000      TETA      D      0   TRAILING ETA (7/9 1.024)
1859 0000      TTI       D      0   TRAILING TI (7/9 1.024)
185A 0000      TVEL      D      0   TRAILING VELOCITY (9/7)
185B 0000      TX        D      0   TABLE INDEX
185C 0000      VBVL      D      0
185D 0000      VEHI      D      0   FIRST VEHICLE FLAG
185E 0000      ZERO      D      0   ZERO FOR DIVIDE
185F 0000      VLNGTH    D      0   VEHICLE LENGTH IN MM
1860 0000      MWCS      D      0   STATUS OF SIGN
1861 0000      MWON      D      0   TIME SIGN ON +5 SECONDS
1862 06D6      MWCONTM   D      X'6D6'   MWC SIGN ON TIME (3.5 SECS)
1863 0000      SPSN      D      0   CURRENT STATUS OF SPEED SIGN
1864 003E      HISP      D      62    43MPH
1865 0038      MDSP      D      56    38MPH
1866 0000      VGRAM     D      0
1867 0000      LSH       D      0
1868 01F4      P500      D      500
               *
               *--
               *
1869           R7TOG     RES    1   R6B BACK-UP CHECK TOGGLE
186A 2710      MAXREDQ   D      10000  20 SEC, .002 SEC COUNTS
186B           RCLR      RES    1    RAMP CLEAR FLAG
```

```
                 166 ' GBS - SG BAND ETA TABLE  #5
186C 249F        167 VMRGTBL  D       9375  LTA      37 FT/SEC
186D 25C6        168          D       9670  TTA
186E 2373        169          D       9075           38
186F 251C        170          D       9500
1870 230F        171          D       8975           39
1871 2481        172          D       9345
1872 2260        173          D       8800           40
1873 23EB        174          D       9195
1874 21E3        175          D       8675           41
1875 235A        176          D       9050
1876 2166        177          D       8550           42
1877 22E7        178          D       8935
1878 2102        179          D       8450           43
1879 226F        180          D       8815
187A 2085        181          D       8325           44
187B 2201        182          D       8705
187C 203A        183          D       8250           45
187D 2198        184          D       8600
187E 1FD6        185          D       8150           46
187F 2139        186          D       8505
1880 1F8B        187          D       8075           47
1881 20DF        188          D       8415
1882 1F40        189          D       8000           48
1883 2085        190          D       8325
1884 1EF5        191          D       7925           49
1885 203A        192          D       8250
1886 1EC3        193          D       7875           50
1887 1FEF        194          D       8175
1888 1E91        195          D       7825           51
1889 1FBD        196          D       8125
188A 1E46        197          D       7750           52
188B 1F8B        198          D       8075
188C 1E2D        199          D       7725           53
188D 1F4A        200          D       8010
188E 1DFB        201          D       7675           54
188F 1F22        202          D       7970
1890 1DDD        203          D       7645           55
1891 1EFA        204          D       7930
1892 1DB5        205          D       7605           56
1893 1ED2        206          D       7890
1894 1D97        207          D       7575           57
1895 1EAF        208          D       7855
1896 1D83        209          D       7555           58
1897 1E91        210          D       7825
1898 1D6A        211          D       7530           59
1899 1E78        212          D       7800
189A 1D5B        213          D       7515           60
189B 1E69        214          D       7785
189C 1D4C        215          D       7500           61
189D 1E55        216          D       7765
189E 1D42        217          D       7490           62
189F 1E46        218          D       7750
18A0 1D3D        219          D       7485           63
18A1 1E3C        220          D       7740
18A2 1D38        221          D       7480           64
18A3 1E32        222          D       7730
```

GBS - SG BAND ETA TABLE #5 PAGE 6

```
18A4 1D38   223            D      7480          65
18A5 1E2D   224            D      7725
18A6 1D38   225            D      7480          66
18A7 1E28   226            D      7720
18A8 0000   227 MTAB       HLT       BRANCH TABLE FOR MODE CONTROL
18A9 11A6   228            JMP    SMMODE
18AA 11BA   229            JMP    SGMODE
18AB 120E   230            JMP    MMMODE
            231
```

GBS INIT ROUTINE WITH START-UP HALT #6 PAGE 7

```
            232 'GBS INIT ROUTINE WITH START-UP HALT  #6
18AC 0000   233            D      0
18AD 60AC   234 INI        STX    $-1  INITIALIZATION ROUTINE
18AE 08C0   235            SSO       REINITIALIZATION?
18AF 10B1   236            JMP    $+2
18B0 10C5   237            JMP    REINIT   YES
18B1 0100   238            CLR       NO. CLEAR
18B2 700B   239            STW    TNMSCK   TEN MSEC CLOCK
18B3 700A   240            STW    MINCLK   ONE MINUTE CLOCK
18B4 7003   241            STW    Q   CURRENT 3 MIN  VOLUME
18B5 7006   242            STW    VBAR   CURRENT 3 MIN AVG VELOCITY
18B6 0088   243            SMB    BIXR
18B7 7001   244            STW    BIXR   BEG INDEX TO VEL TABLE
18B8 0088   245            SMB    CIXR
18B9 7002   246            STW    CIXR   CURRENT INDEX TO VEL TABLE
18BA 0088   247            SMB    TVL
18BB 7003   248            STW    TVL
18BC 0083   249            SMB    NUMV   NUMBER OF VEHICLES IN
18BD 7004   250            STW    NUMV   VELOCITY TOTAL
18BE 7010   251            STW    SSFLAG   3 MINUTE FLAG (DO NOT USE Q)
18BF 8757   252            LDW    =44   SM TO SG AT 30 MPH
18C0 700E   253            STW    SMSGBD
18C1 8758   254            LDW    =59   SG TO MM AT 40 MPH
18C2 700F   255            STW    SGMMBD
18C3 0000   256            HLT
18C4 15A0   257            JMP    EXIM
            258
```

```
                ' GBS SYSTEM RE-INITIALIZE ROUTINE  #7
18C5 0100       REINIT   CLR
18C6 7063                STW   SPSN    CLEAR SPEED SIGN
18C7 7061                STW   MWON    CLEAR MERGE ON TIMER
18C8 7060                STW   MWCS    MERGE WITH CAUTION SIGN OFF
18C9 7024                STW   INTREDFL   CLEAR INIT TO RED FLAG
18CA 7053                STW   REDFLAG
18CB 7005                STW   VMRG
18CC 7002                STW   MODE
18CD 7025                STW   VWAITFLG
18CE 7069                STW   R7TOG
18CF 706B                STW   RCLR
18D0 7616                STW   MCLK
18D1 7617                STW   MCNT
18D2 7603                STW   RCLK
18D3 7604                STW   RCNT
18D4 712F                STW   VWCLK
18D5 7130                STW   VWCNT
18D6 7105                STW   SCLK
18D7 7106                STW   SCNT
18D8 9759                LDX   =-10
18D9 0082                SMB   DBUF    CLEAR DD
18DA 7833                STW * DBUF+10
18DB 0401                IXS   1
18DC 10D9                JMP   $-3
18DD 0082                SMB   RLNBND
18DE 7036                STW   RLNBND
18DF 975A                LDX   =-5
18E0 0084                SMB   SBUF    CLEAR SD
18E1 78CD                STW * SBUF+5
18E2 0401                IXS   1
18E3 10E0                JMP   $-3
18E4 875B                LDW   =1
18E5 7014                STW   GRNOPTFL
18E6 875C                LDW   =-1     SET INIT FLAG AND FLASHING
18E7 7004                STW   INFG    AMBER LIGHT ON-OFF INDICATOR
18E8 7029                STW   FAFG
18E9 0080                SMB   TIME    STORE TIME BEGAN INIT AND
18EA 8021                LDW   TIME    TIME LIGHT WAS TURNED AMBER
18EB 7012                STW   ARTI
18EC 7027                STW   FAON
18ED 7630                STW   MAS
18EE 762F                STW   RMPS
18EF 90AC                LDX   INI-1
18F0 2800                JSX * 0   RETURN
```

```
                 ' GBS ... MAIN ENTRY POINT  #8
      18F1       GBS        EQU    $
  18F1 25A3                 JSX    SSCK   SET OVERHEAD WINKOMATIC SIGN
  18F2 25B5                 JSX    MWC    SEE IF MWC SIGN SHOULD BE ON
  18F3 25E7                 JSX    CHKKI9
  18F4 0089                 SMB    GBSSTRT
  18F5 247C                 JSX    GBSSTRT   COPY GBT ETC FROM GBUD
                 *
                 *-
                 *
  18F6 02F5                 DIN    15,5   RAMP CLEAR CHECK
  18F7 0A15                 SLL    5        R6B
  18F8 0820                 SAM               ON
  18F9 1103                 JMP    RSCLR      OFF
  18FA 0A15                 SLL    5        R7
  18FB 0820                 SAM               ON
  18FC 1103                 JMP    RSCLR      OFF
  18FD 0080                 SMB    TIME
  18FE 8021                 LDW    TIME
  18FF 7105                 STW    SCLK
  1900 0100                 CLR
  1901 7106                 STW    SCNT
  1902 1109                 JMP    SCNTO
      1903       RSCLR      EQU    $
  1903 008F                 CUSC
S 1904 26E8
  1905 0000      SCLK       D      0
  1906 0000      SCNT       D      0
  1907 000A      SLIM       D      10  SEC
  1908 110C                 JMP    SCOVR   TIME OUT, RAMP CLEAR
      1909       SCNTO      EQU    $
  1909 875B                 LDW    =1   NO TIME OUT,
  190A 706B                 STW    RCLR  FLAG NOT CLEAR
  190B 1114                 JMP    SCAST
      190C       SCOVR      EQU    $
  190C 0100                 CLR          TIME OUT
  190D 706B                 STW    RCLR  FLAG CLEAR
  190E 0080                 SMB    TIME  KEEP TIMER
  190F 8021                 LDW    TIME  OVERFLOWING
  1910 B75D                 SUB    =500
  1911 7105                 STW    SCLK
  1912 8107                 LDW    SLIM
  1913 7106                 STW    SCNT
      1914       SCAST      EQU    $
                 *
  1914 02F5                 DIN    15,5  R7 BACK-UP OF R6B
  1915 E629                 AND    TSR6A  R6A ACT?
  1916 0800                 SAZ
  1917 1119                 JMP    $+2
  1918 111B                 JMP    R7CHK     NO, CHECK R7
  1919 0100                 CLR              YES, CLEAR R7 TOGGLE
  191A 7069                 STW    R7TOG
      191B       R7CHK      EQU    $
  191B 8069                 LDW    R7TOG   CHECK R7 TOGGLE
  191C 0810                 SAP
  191D 113C                 JMP    R7TEXT  -VE, ALREADY SET
  191E 0800                 SAZ
```

```
  191F 112D              JMP   R7TOGP  +VE, CHECK HOW LONG
  1920 02F5              DIN   15,5    0, R7 CHECK
  1921 E62E              AND   TSR7     ACTIVE?
  1922 0800              SAZ
  1923 1125              JMP   $+2       YES
  1924 113C              JMP   R7TEXT    NO
  1925 0100              CLR
  1926 7130              STW   VWCNT   ZERO TIME COUNT
  1927 0080              SMB   TIME    RESET CLOCK
  1928 8021              LDW   TIME
  1929 712F              STW   VWCLK
  192A 875B              LDW   =1      SET R7 TOGGLE +VE
  192B 7069              STW   R7TOG
  192C 113C              JMP   R7TEXT
       192D    R7TOGP    EQU   $       TOGGLE .GT. 0
  192D 008F              CUSC
S 192E 26E8
               *                 R7 COUNTING
  192F 0000    VWCLK     D     0       WAITING FOR
  1930 0000    VWCNT     D     0       R6A
  1931 000F    VWLIM     D     15  SEC
  1932 1134              JMP   $+2     TIME OUT, LET IT GO
  1933 113C              JMP   R7TEXT    STILL COUNTING
  1934 875C              LDW   =-1    SET TOGGLE FOR R7 TIME OUT
  1935 7069              STW   R7TOG
  1936 8131              LDW   VWLIM
  1937 7130              STW   VWCNT   KEEP TIMER OVERFLOWING
  1938 0080              SMB   TIME
  1939 8021              LDW   TIME
  193A B75D              SUB   =500
  193B 712F              STW   VWCLK
       193C    R7TEXT    EQU   $
               *
               *-
               *
  193C 800B              LDW   TNMSCK    INCREMENT 10 MSEC CLOCK
  193D A75B              ADD   =1
  193E 700B              STW   TNMSCK
  193F F75E              CMW   =X'17D4'   ONE MINUTE (60000 MSECS)?
  1940 0860              SEQ
  1941 1150              JMP   BGBS   NO
  1942 0100              CLR     YES. CLEAR TEN MSEC CLOCK
  1943 700B              STW   TNMSCK
  1944 800A              LDW   MINCLK   BUMP ONE MINUTE CLOCK BY 1
  1945 A75B              ADD   =1
  1946 700A              STW   MINCLK
  1947 F75F              CMW   =3   3 MIN. SINCE SYSTEM START UP?
  1948 0860              SEQ
  1949 1150              JMP   BGBS   NO
  194A 875C              LDW   =-1  YES.  SET 3 MIN. FLAG
  194B 7010              STW   SSFLAG
  194C 8760              LDW   =41
  194D 700E              STW   SMSGBD
  194E 8761              LDW   =56
  194F 700F              STW   SGMMBD
```

```
              ' BGBS ROUTINE  #9
1950 0088     BGBS      SMB   JVELVL
1951 2000               JSX   JVELVL
1952 8004               LDW   INFG
1953 0820               SAM
1954 1172               JMP   FINT
1955 8012               LDW   ARTI
1956 A762               ADD   =X'1D4C'   *15 SEC RE-INIT TIME *****
1957 7056               STW   TMTM
1958 0080               SMB   TIME
1959 8021               LDW   TIME
195A B056               SUB   TMTM
195B 0810               SAP
195C 1168               JMP   INONOF
195D 0088               SMB   NUMV   YES
195E 8004               LDW   NUMV    ANY VEH CROSSED F1 SINCE
195F 0800               SAZ      SYSTEM START UP?
1960 1162               JMP   $+2
1961 1168               JMP   INONOF
1962 0100               CLR          YES.CLEAR
1963 7004               STW   INFG   INITIALIZATION FLAG
1964 7013               STW   EXPFLG   AND EXPRESS FLAG
1965 875B               LDW   =1   SET MODE TO SM TO GIVE STARTING
1966 7002               STW   MODE   POINT FOR MODE SWITCHING LOGIC
1967 1172               JMP   FINT
1968 23B4     INONOF    JSX   ONOF   DETERMINE STATE OF FLASHING
              *                      AMBER LIGHTS
1969 8029               LDW   FAFG   LIGHTS ON?
196A 0810               SAP
196B 116F               JMP   INTON   YES
196C 0100               CLR           NO
196D 7015               STW   LIGHT   TURN LIGHT OFF
196E 1192               JMP   OUTPUT
196F 8763     INTON     LDW   =X'20'   TURN LIGHT AMBER
1970 7015               STW   LIGHT
1971 1192               JMP   OUTPUT
```

```
                ' JMPMODE AND OUTPUT ROUTINES  #10
1972 875B       FINT      LDW    =1   SET FLAG SO THAT DD AND SD
1973 0089                 SMB    ALLG   ARE NOT SET ALL GREEN
1974 7479                 STW    ALLG
1975 08E0                 SS2       OPERATOR SELECTED GREEN?
1976 117D                 JMP    GRNOPTST   YES
1977 8014                 LDW    GRNOPTFL   NO.
1978 F75B                 CMW    =1
1979 0860                 SEQ
197A 117D                 JMP    GRNOPTST  `NO
197B 0100                 CLR       YES.
197C 7014                 STW    GRNOPTFL
197D 08E0       GRNOPTST  SS2
197E 1182                 JMP    $+4
197F 9002       JMPMODE   LDX    MODE   NO
1980 0040                 SLM
1981 18A8                 JMP *  MTAB   JUMP TO APPROPRIATE MODE
1982 8014                 LDW    GRNOPTFL   YES.
1983 0800                 SAZ
1984 117F                 JMP    JMPMODE
1985 875C                 LDW    =-1
1986 7014                 STW    GRNOPTFL
1987 8764                 LDW    =X'10'
1988 7015                 STW    LIGHT
1989 117F                 JMP    JMPMODE
198A 875F       MM        LDW    =3
198B 7002                 STW    MODE   SET MODE TO MM
198C 235E                 JSX    ALLCLR   CHECK IF ANY VEHICLES ON HWY
198D 0100                 CLR       YES. CLEAR SENSOR ACTIVE FLAG
198E 7009                 STW    ACTF
198F 23D7                 JSX    BINK   CHECK BINS
1990 8764                 LDW    =X'10'   TURN LIGHT GREEN
1991 7015                 STW    LIGHT
     1992       OUTPUT    EQU    $   ***PATCH 1 INSERTED***
1992 8002                 LDW    MODE
1993 F75B                 CMW    =1
1994 0870                 SNE
1995 1199                 JMP    PATCH1A
1996 8060                 LDW    MWCS     MODE WAS NOT 1
1997 E765                 AND    =-2
1998 119B                 JMP    PATCH1B
1999 8060       PATCH1A   LDW    MWCS     MODE WAS 1
199A C75B                 ORI    =1
199B A015       PATCH1B   ADD    LIGHT
199C 0A10       SETSM     NOP
199D A063                 ADD    SPSN   CURRENT SPEED SIGN SETTING
199E 0082                 SMB    DBUF
199F 7033                 STW    DBUF+10  RAMP SIGN CONTROL FOR SCU
19A0 0084                 SMB    DSTA
19A1 70C0                 STW    DSTA+1   RAMP SIGN CONTROL FOR SD
19A2 0089                 SMB    GBSDONE
19A3 248F                 JSX    GBSDONE   COPY NEW GBT TO GBUD
19A4 0080                 SMB    EXEC
19A5 1023                 JMP    EXEC     RETURN TO EXECUTIVE
```

```
          ' SMMODE ROUTINE   #11
19A6 8006   SMMODE   LDW   VBAR
19A7 F00E            CMW   SMSGBD   VBAR.GT.28(30) M.P.H.
19A8 0880            SGR
19A9 11B1            JMP   SM   NO.  MODE REMAINS SM
19AA 0100            CLR      YES. CLEAR SG  MM FIRST TIME
19AB 7023            STW   SGMMFG   THROUGH FLAG
19AC 875C            LDW   =-1   NO.
19AD 7024            STW   INTREDFL
19AE 8766            LDW   =2   SET MODE TO SG
19AF 7002            STW   MODE
19B0 1192            JMP   OUTPUT
19B1 875B   SM       LDW   =1
19B2 7002            STW   MODE   KEEP MODE SM
19B3 902F            LDX   K19QSM
19B4 801B            LDW   MAXRED
19B5 F02D            CMW   K19QSG
19B6 0860            SEQ
19B7 9030            LDX   K19NOQSM
19B8 601C            STX   MINRED
19B9 1243            JMP   SMANDSG
```

```
                ' SGMODE ROUTINE  #12
19BA 8023       SGMODE   LDW   SGMMFG   GOING FROM SG TO MM?
19BB 0800                SAZ
19BC 11F0                JMP   TWRED   YES
19BD 8006                LDW   VBAR   NO
19BE F767                CMW   =37   VBAR LESS THAN 25 M.P.H?
19BF 0840                SLS
19C0 11C9                JMP   TSGMM   NO
19C1 0100                CLR     YES. KILL ANY ACCELERATING BANDS
19C2 0089                SMB   NBND
19C3 7400                STW   NBND
19C4 875C                LDW   =-1
19C5 7024                STW   INTREDFL
19C6 875B                LDW   =1   CODE FOR SM
19C7 7002                STW   MODE
19C8 1192                JMP   OUTPUT
19C9 F00F       TSGMM    CMW   SGMMBD   VBAR .GT. 38 (40) M.P.H.
19CA 0880                SGR
19CB 1207                JMP   SG   NO. MOD  REMAINS SG
19CC 8010                LDW   SSFLAG   3 MINUTES SINCE INIT?
19CD 0820                SAM
19CE 11D3                JMP   $+5   NO. DO NOT USE VOLUME (Q) TEST
19CF 8003                LDW   Q   YES. Q .LT. QSGMM VEH PER HR?
19D0 F03F                CMW   QSGMM
19D1 0840                SLS
19D2 1207                JMP   SG   NO. MODE REMAINS SG
19D3 8013                LDW   EXPFLG   YES. CAME FROM INIT?
19D4 0820                SAM
19D5 11FB                JMP   SETMM   YES
19D6 8015                LDW   LIGHT   NO
19D7 F768                CMW   =X'40'   LIGHT RED?
19D8 0860                SEQ
19D9 1207                JMP   SG   NO. MODE REMAINS SG
19DA 8018                LDW   TIMERED   YES
19DB A769                ADD   =X'3E8'
19DC 7056                STW   TMTM
19DD 0080                SMB   TIME
19DE 8021                LDW   TIME
19DF B056                SUB   TMTM   RED 2 SECONDS?
19E0 0810                SAP
19E1 1237                JMP   FLT   NO. JUST FILTER
19E2 8023                LDW   SGMMFG   FIRST TIME THRU SG TO  MM?
19E3 0800                SAZ
19E4 11F0                JMP   TWRED   NO
```

```
               ' SGMODE ROUTINE - CONTINUED  #13
19E5 0100                    CLR        YES. KILL ANY ACCELERATING BANDS
19E6 0089                    SMB     NBND
19E7 7400                    STW     NBND
19E8 8764                    LDW     =X'10'    YES. SET LIGHT GREEN
19E9 7015                    STW     LIGHT
19EA 0080                    SMB     TIME   STORE TIME SET GREEN
19EB 8021                    LDW     TIME
19EC 7022                    STW     SGMMT
19ED 875C                    LDW     =-1   SET SG  MM FIRST TIME THRU FLAG
19EE 7023                    STW     SGMMFG
19EF 1192                    JMP     OUTPUT
19F0 801E      TWRED         LDW     WREDFLAG   LAST RED A 'WAIT IN RED'?
19F1 0800                    SAZ
19F2 11FB                    JMP     SETMM   YES
19F3 8022                    LDW     SGMMT   NO
19F4 A76A                    ADD     =X'1F40'
19F5 7056                    STW     TMTM
19F6 0080                    SMB     TIME
19F7 8021                    LDW     TIME
19F8 B056                    SUB     TMTM   WAITING IN SG  MM FOR 16 SEC?
19F9 0810                    SAP
19FA 1192                    JMP     OUTPUT   NO.
               *-
     19FB      SETMM         EQU     $
19FB 806B                    LDW     RCLR   RAMP CLEAR?
19FC 0800                    SAZ
19FD 11FF                    JMP     $+2   NO, DON'T CHANGE YET
19FE 1202                    JMP     CTMM  YES, LET IT GO
19FF 875C                    LDW     =-1   KEEP IT COMING TO SETMM
1A00 701E                    STW     WREDFLAG
1A01 1192                    JMP     OUTPUT
     1A02      CTMM          EQU     $    LET IT GO
               *-
1A02 0100                    CLR        YES, CLEAR MM TO SG FIRST TIME
1A03 7020                    STW     MMSGFG   THROUGH FLAG
1A04 875F                    LDW     =3   SET MODE TO MM
1A05 7002                    STW     MODE
1A06 1192                    JMP     OUTPUT
```

```
               ' SG AND MMMODE ROUTINES  #14
1A07 8766        SG       LDW    =2
1A08 7002                 STW    MODE   KEEP MODE SG
1A09 801D                 LDW    SGMINRED   SET MIN. RED WAIT TIME
1A0A 701C                 STW    MINRED   TO 2 SECONDS
1A0B 802E                 LDW    K19NOQSG
1A0C 701F                 STW    WREDTIME
1A0D 1243                 JMP    SMANDSG
1A0E 8020        MMMODE   LDW    MMSGFG  TEST FIRST TIME THRU FLAG
1A0F 0800                 SAZ
1A10 1229                 JMP    STSG
1A11 8006                 LDW    VBAR
1A12 F76B                 CMW    =51    VBAR = 35 M.P.H?
1A13 0880                 SGR
1A14 121C                 JMP    MMSG   NO
1A15 8010                 LDW    SSFLAG   3 MIN. SINCE SYSTEM START UP
1A16 0820                 SAM
1A17 118A                 JMP    MM   NO. MODE REMAINS MM
1A18 8003                 LDW    Q   YES. Q .LT. QMMSG VEH PER HR?
1A19 F04D                 CMW    QMMSG
1A1A 0880                 SGR
1A1B 118A                 JMP    MM   NO. MODE REMAINS MM
1A1C 8763        MMSG     LDW    =X'20'   YES. TURN LIGHT AMBER
1A1D 7015                 STW    LIGHT
1A1E 0080                 SMB    TIME   STORE TIME SET AMBER
1A1F 8021                 LDW    TIME
1A20 7021                 STW    MMSGAM
1A21 875C                 LDW    =-1   SET MM  SG FIRST TIME THRU FLAG
1A22 7020                 STW    MMSGFG
1A23 0100                 CLR       KILL ANY CONSTANT VELOCITY BANDS
1A24 0089                 SMB    NBND
1A25 7400                 STW    NBND
1A26 1192                 JMP    OUTPUT
1A27 0A10                 NOP       OLD CODE HERE KILLED SPEED
1A28 0A10                 NOP       SIGN FOR MM TO SG TRANSFER
1A29 8021        STSG     LDW    MMSGAM
1A2A A76C                 ADD    =X'6D6'
1A2B 7056                 STW    TMTM
1A2C 0080                 SMB    TIME
1A2D 8021                 LDW    TIME
1A2E B056                 SUB    TMTM   AMBER 3.5 SECONDS?
1A2F 0810                 SAP
1A30 1192                 JMP    OUTPUT   NO.
1A31 0100                 CLR       YES
1A32 7024                 STW    INTREDFL   CLEAR INIT TO RED FLAG
1A33 7023                 STW    SGMMFG
1A34 8766                 LDW    =2         SET MODE TO SG
1A35 7002                 STW    MODE
1A36 1192                 JMP    OUTPUT
```

```
                ' FLT ROUTINE AND FILT SUBROUTINE  #15
1A37 223A       FLT      JSX   FILT
1A38 1192                JMP   OUTPUT
1A39 0000                D     0
1A3A 6239       FILT     STX   $-1  STORE INDEX
1A3B 8009                LDW   ACTF   ANY HIGHWAY SENSOR ACTIVATED?
1A3C 0830                SAO
1A3D 1241                JMP   RTFILT   NO. RETURN
1A3E 236F                JSX   FILTER   YES. DROP ACCELERATING BANDS
1A3F 0100                CLR
1A40 7009                STW   ACTF   YES. CLEAR SENSOR ACTIVE FLAG
1A41 9239       RTFILT   LDX   FILT-1
1A42 2800                JSX * 0   RETURN
```

```
                   ' SMANDSG ROUTINE  #16
1A43 9002          SMANDSG  LDX   MODE     MODE SG?
1A44 0850                   SXE
1A45 1247                   JMP   $+2
1A46 223A                   JSX   FILT
1A47 8024                   LDW   INTREDFL   NO. INITIALIZE TO RED?
1A48 0810                   SAP
1A49 1253                   JMP   THRUBFR   NO
1A4A 875C                   LDW   =-1   YES
1A4B 7024                   STW   INTREDFL
1A4C 7013                   STW   EXPFLG
1A4D 8768          SETRED   LDW   =X'40'   TURN LIGHT RED
1A4E 7015                   STW   LIGHT
1A4F 0080                   SMB   TIME
1A50 8021                   LDW   TIME
1A51 7018                   STW   TIMERED   STORE TIME SET RED
1A52 1192                   JMP   OUTPUT
1A53 8015          THRUBFR  LDW   LIGHT   LIGHT GREEN?
1A54 F764                   CMW   =X'10'
1A55 0870                   SNE
1A56 12E9                   JMP   CHECKGRN   YES
1A57 F763                   CMW   =X'20'   NO. LIGHT AMBER?
1A58 0860                   SEQ
1A59 1263                   JMP   STATERED   NO
1A5A 8017                   LDW   TIMEAMBR   YES
1A5B A01A                   ADD   MAXAMBR
1A5C 7056                   STW   TMTM
1A5D 0080                   SMB   TIME
1A5E 8021                   LDW   TIME
1A5F B056                   SUB   TMTM   AMBER 1 SECOND?
1A60 0810                   SAP
1A61 1192                   JMP   OUTPUT   NO.
1A62 124D                   JMP   SETRED   YES.
1A63 02F5          STATERED DIN   X'F',5
1A64 E629                   AND   TSR6A   TAMPA CHECKOUT SENSOR MASK
1A65 0800                   SAZ
1A66 1192                   JMP   OUTPUT   YES
1A67 8053                   LDW   REDFLAG   NO. MINIMUM RED (2 SECONDS)
1A68 0810                   SAP                 SATISFIED?
1A69 1274                   JMP   CKDEMAND   YES
1A6A 8018                   LDW   TIMERED   NO
1A6B A01C                   ADD   MINRED
1A6C 7056                   STW   TMTM
1A6D 0080                   SMB   TIME
1A6E 8021                   LDW   TIME
1A6F B056                   SUB   TMTM   RED 2 (SG) OR 5 OR 10 (SM)SEC?
1A70 0810                   SAP
1A71 1192                   JMP   OUTPUT   NO.
1A72 875C                   LDW   =-1   YES. SET MINIMUM RED FLAG
1A73 7053                   STW   REDFLAG
```

```
               ' SMANDSG ROUTINE - CONTINUED   #17
1A74 02F5      CKDEMAND DIN   X'F',5
1A75 E62A               AND   TSR6B
1A76 0800               SAZ      VEHICLE WAITING?
1A77 127E               JMP   VWAITING   YES
               *-
1A78 8069               LDW   R7TOG   R7 TIMED OUT?
1A79 0810               SAP
1A7A 127E               JMP   VWAITING    YES, RELEASE VEHICLE
               *-
1A7B 0100               CLR      NO. CLEAR VEHICLE WAIT FLAG
1A7C 7025               STW   VWAITFLG
1A7D 1192               JMP   OUTPUT
1A7E 8002      VWAITING LDW   MODE   MODE SM?
1A7F 0830               SAO
1A80 129A               JMP   MODESG   NO. MODE IS SG
1A81 02F5      SETGREEN DIN   X'F',5   YES
1A82 E629               AND   TSR6A
1A83 0800               SAZ    PASSAGE LOOP ACTIVATED?
1A84 1192               JMP   OUTPUT   YES.
1A85 8764               LDW   =X'10'   NO.
1A86 7015               STW   LIGHT
1A87 0080               SMB   TIME
1A88 8021               LDW   TIME
1A89 7016               STW   TIMEGRN
1A8A 0100               CLR
1A8B 7053               STW   REDFLAG
1A8C 8018               LDW   TIMERED
1A8D A01F               ADD   WREDTIME
1A8E 7056               STW   TMTM
1A8F 0080               SMB   TIME
1A90 8021               LDW   TIME
1A91 B056               SUB   TMTM   RED 16 SEC ('WAIT IN RED')?
1A92 0810               SAP
1A93 1297               JMP   $+4
1A94 875C               LDW   =-1   YES. SET 'WAIT IN RED' FLAG
1A95 701E               STW   WREDFLAG
1A96 1192               JMP   OUTPUT
1A97 0100               CLR      NO. CLEAR 'WAIT IN RED' FLAG
1A98 701E               STW   WREDFLAG
1A99 1192               JMP   OUTPUT
```

```
               ' MODESG ROUTINE  #18
1A9A 8025        MODESG    LDW   VWAITFLG   1ST TIME SAW VEH WAITING?
1A9B 0810                  SAP
1A9C 12A2                  JMP   MODESGBF   NO
1A9D 875C                  LDW   =-1   YES. SET VEHICLE WAIT FLAG
1A9E 7025                  STW   VWAITFLG
1A9F 0080                  SMB   TIME   STORE TIME VEH STARTED WAIT
1AA0 8021                  LDW   TIME
1AA1 7026                  STW   VWAITIME
1AA2 8005        MODESGBF  LDW   VMRG   GET SG MODE BAND ETA VALUES
1AA3 B767                  SUB   =37
1AA4 0911                  SLA   1
1AA5 0130                  CAX  .
1AA6 0040                  SLM
1AA7 886C                  LDW * VMRGTBL   LT..
1AA8 0080                  SMB   TIME
1AA9 A021                  ADD   TIME
1AAA 7040                  STW   LTA
1AAB B04F                  SUB   LHDY
1AAC B044                  SUB   LHDYTLLN
1AAD 7042                  STW   LTAMLHDY
1AAE 886D                  LDW * VMRGTBL+1   TTA
1AAF 0080                  SMB   TIME
1AB0 A021                  ADD   TIME
1AB1 7041                  STW   TTA
1AB2 A050                  ADD   THDY
1AB3 A045                  ADD   THDYTLLN
1AB4 7043                  STW   TTAPTHDY
1AB5 8048                  LDW   RGPTOLLN
1AB6 704A                  STW   RGAPTOL
1AB7 2304                  JSX   LNDL   CAN BAND BE LAUNCHED?
1AB8 0040                  SLM
1AB9 803E                  LDW   LDFG   LAUNCH BAND?
1ABA 0810                  SAP
1ABB 12DF                  JMP   NOKL   NO
1ABC 0089                  SMB   NBND   YES.
1ABD 8400                  LDW   NBND
1ABE 0A13                  SLL   3
1ABF 0130                  CAX
1AC0 0089                  SMB   NBND   BUMP BAND COUNT
1AC1 8400                  LDW   NBND
1AC2 A75B                  ADD   =1
1AC3 E00C                  AND   K51F   KILL ALL BANDS IF K51 ON
1AC4 0089                  SMB   NBND
1AC5 7400                  STW   NBND
```

```
                ' BAND LAUNCH - SG MODE  #19
1AC6 0080                      SMB    TIME    SET TIME BAND LAUNCHED
1AC7 8021                      LDW    TIME
1AC8 0039                      SMB    GBT
1AC9 7C01                      STW  * GBT
1ACA 876D                      LDW    =X'8000'
1ACB 0089                      SMB    GBT
1ACC 7C02                      STW  * GBT+1   ACC FLAG 'ON' AT ZERO VEL
1ACD 876E                      LDW    =276    TAMPA VALUE (SGXLE)*********
1ACE 0089                      SMB    GBT
1ACF 7C03                      STW  * GBT+2
1AD0 876F                      LDW    =244    TAMPA VALUE (SGXTE)*********
1AD1 0089                      SMB    GBT
1AD2 7C05                      STW  * GBT+4
1AD3 0100                      CLR
1AD4 0089                      SMB    GBT
1AD5 7C04                      STW  * GBT+3    CLR FRAC LEADING EDGE
1AD6 0089                      SMB    GBT
1AD7 7C06                      STW  * GBT+5    CLR FRAC TRAILING EDGE
1AD8 8040                      LDW    LTA
1AD9 0089                      SMB    GBT
1ADA 7C07                      STW  * GBT+6    BAND LEAD EDGE ETA
1ADB 8041                      LDW    TTA
1ADC 0089                      SMB    GBT
1ADD 7C08                      STW  * GBT+7    BAND TRAIL EDGE ETA
1ADE 1281                      JMP    SETGREEN    TURN LIGHT GREEN
1ADF 8026       NOKL           LDW    VWAITIME
1AE0 A01B                      ADD    MAXRED
1AE1 B01C                      SUB    MINRED
1AE2 7056                      STW    TMTM
1AE3 0080                      SMB    TIME
1AE4 8021                      LDW    TIME
1AE5 B056                      SUB    TMTM    RED 8 OR 4 (IF QUEUE) SEC?
1AE6 0810                      SAP
1AE7 1192                      JMP    OUTPUT   NO.
1AE8 1281                      JMP    SETGREEN   YES. TURN LIGHT GREEN
1AE9 02F5       CHECKGRN       DIN    X'F',5
1AEA E629                      AND    TSR6A
1AEB 0800                      SAZ       CHECKOUT LOOP ACTIVATED?
1AEC 12FE                      JMP    OPTFLG   YES
1AED 8014                      LDW    GRNOPTFL   NO. OPERATOR OPTION GREEN?
1AEE 0810                      SAP
1AEF 1192                      JMP    OUTPUT   YES.
1AF0 8016                      LDW    TIMEGRN   NO
1AF1 A019                      ADD    MAXGREEN
1AF2 7056                      STW    TMTM
1AF3 0080                      SMB    TIME
1AF4 8021                      LDW    TIME
1AF5 B056                      SUB    TMTM   GREEN 5 SECONDS?
1AF6 0810                      SAP
1AF7 1192                      JMP    OUTPUT   NO.
```

```
                ' SETAMBER ROUTINE  #20
1AF8 8763       SETAMBER LDW    =X'20'   YES. TURN LIGHT AMBER
1AF9 7015                STW    LIGHT
1AFA 0080                SMB    TIME
1AFB 8021                LDW    TIME
1AFC 7017                STW    TIMEAMBR   STORE TIME SET AMBER
1AFD 1192                JMP    OUTPUT
1AFE 875B       OPTFLG   LDW    =1         SETFLAG TO 1 TO INDICATE
1AFF 7014                STW    GRNOPTFL    PASSAGE LOOP ACTIVATED
1B00 0100                CLR               CLEAR VEHICLE WAIT FLAG
1B01 7025                STW    VWAITFLG
1B02 12F8                JMP    SETAMBER
```

```
                883 ' LNDL ROUTINE  #21
1B03 0000       884             D      0
1B04 6303       885 LNDL        STX    $-1  STORE INDEX
1B05 0040       886             SLM
1B06 0100       887             CLR       CLEAR
1B07 703E       888             STW    LDFG   LAUNCH-DELETE FLAG
1B08 704C       889             STW    LVEHFLAG   AND LEADING VEHICLE FLAG
                890 *-
                891 * PATCH 2 INSERTED***
                892 *
1B09 0080       893             SMB    TIME
1B0A 8021       894             LDW    TIME
1B0B 7039       895             STW    LETA
1B0C 9770       896             LDX    =-14
                897 *-
1B0D 6034       898 BNK         STX    SIXR
1B0E 0082       899             SMB    PNTW
1B0F 8BD6       900             LDW *  PNTW+14
1B10 702A       901             STW    BEGP   BEGINNING POINTER
1B11 0082       902             SMB    PNTW
1B12 8BD7       903             LDW *  PNTW+15
1B13 702B       904             STW    ENDP   END POINTER
1B14 F02A       905             CMW    BEGP   THIS HIGHWAY LIST EMPTY?
1B15 0870       906             SNE
1B16 1359       907             JMP    BEM   YES. CHECK NEXT BIN
1B17 902A       908 NVEH        LDX    BEGP   NO
1B18 0082       909             SMB    PNTW
1B19 9800       910             LDX *  0
1B1A 8039       911             LDW    LETA   PREVIOUS ETA
1B1B 0082       912             SMB    PNTW
1B1C B800       913             SUB *  0   CURRENT ETA
1B1D 0820       914             SAM       CURRENT ETA .GT. PREVIOUS ETA ?
1B1E 1353       915             JMP    IBP   NO. TRY NEXT VEHICLE
1B1F 0082       916             SMB    PNTW   YES
1B20 8800       917             LDW *  0   CURRENT VEH ARRIVE BEFORE LEADING
1B21 B042       918             SUB    LTAMLHDY   EDGE OF BAND?
1B22 0810       919             SAP
1B23 134F       920             JMP    SPAR   YES
1B24 A042       921             ADD    LTAMLHDY   NO
1B25 0110       922             CMP
1B26 A043       923             ADD    TTAPTHDY
1B27 0820       924             SAM
1B28 134B       925             JMP    NOKF   NO. DELETE BAND
1B29 804C       926             LDW    LVEHFLAG   YES. FOUND LEAD VEH?
1B2A 0820       927             SAM
1B2B 134D       928             JMP    RTLD   NO. DO NOT DELETE BAND
                929
```

```
                ' LNDL ROUTINE - CONTINUED #22
1B2C 902A                 LDX    BEGP   YES
1B2D 0082                 SMB    PNTW   STORE PARAMETERS OF TRAILING
1B2E 9800                 LDX * 0       VEH FOR USE IN GRAM
1B2F 0082                 SMB    PNTW
1B30 8800                 LDW * 0
1B31 7058                 STW    TETA   TRAILING ETA
1B32 0082                 SMB    PNTW
1B33 8801                 LDW * 1
1B34 7059                 STW    TTI  TRAILING TI
1B35 0082                 SMB    PNTW
1B36 8802                 LDW * 2
1B37 705A                 STW    TVEL   TRAILING VELOCITY
1B38 0082                 SMB    PNTW
1B39 8800                 LDW * 0
1B3A B039                 SUB    LETA
1B3B B03A                 SUB    LLEN   MEASURED GAP= TRAIL ETA - LEAD
1B3C 7054                 STW    MGAP   ETA - TRAIL LENGTH
1B3D 8039                 LDW    LETA   STORE PARAM OF LEAD VEH FOR
1B3E 7037                 STW    GLTA   GAP REQUIREMENT TEST.
1B3F 803C                 LDW    LTI
1B40 7038                 STW    GLTI   LEADING TI
1B41 8057                 LDW    SVEL
1B42 7052                 STW    LVEL   LEADING VELOCITY
1B43 24E2                 JSX    GRAM   CALCULATE REQUIRED GAP
1B44 0040                 SLM
1B45 8054                 LDW    MGAP
1B46 B04A                 SUB    RGAPTOL   GAP ALLOWANCE
1B47 B055                 SUB    RGAP   DELETE BAND (RGAP  MGAP)?
1B48 0810                 SAP
1B49 134B                 JMP    NOKF   YES
1B4A 134D                 JMP    RTLD   NO
1B4B 875C       NOKF      LDW    =-1  SET FLAG NOT TO LAUNCH BAND,
1B4C 703E                 STW    LDFG   OR TO DELETE BAND.
1B4D 9303       RTLD      LDX    LNDL-1
1B4E 2800                 JSX * 0   RETURN
1B4F 2499       SPAR      JSX    STPM   STORE VEHICLE PARAMETERS
1B50 0040                 SLM
1B51 875C                 LDW    =-1  SET LEAD VEH FLAG INDICATING
1B52 704C                 STW    LVEHFLAG   THAT THERE IS A LEAD VEH
1B53 802A       IBP       LDW    BEGP   INCREMENT BEGINNING POINTER
1B54 A75B                 ADD    =1     TO REFERENCE
1B55 702A                 STW    BEGP   NEXT VEHICLE
1B56 F02B                 CMW    ENDP   THIS LIST EXHAUSTED?
1B57 0860                 SEQ
1B58 1317                 JMP    NVEH   NO. CHECK NEXT VEHICLE
1B59 9034       BEM       LDX    SIXR   YES
1B5A 0402                 IXS    2
1B5B 130D                 JMP    BNK  NO.
1B5C 134D                 JMP    RTLD   YES. RETURN
```

```
          ' ALLCLR ROUTINE  #23
1B5D 0000            D      0
1B5E 635D  ALLCLR    STX    $-1  STORE INDEX
1B5F 0040            SLM
1B60 9770            LDX    =-14
1B61 0082  ABNK      SMB    PNTW
1B62 8BD6            LDW  * PNTW+14   BEGINNING POINTER
1B63 0082            SMB    PNTW
1B64 FBD7            CMW  * PNTW+15   END PNTER. ANY VEH IN BIN?
1B65 0860            SEQ
1B66 136C            JMP    RTALCR   YES. RETURN TO GBS
1B67 0402            IXS    2        NO. ALL BINS CHECKED?
1B68 1361            JMP    ABNK   NO. BUMP INDEX AND CONTINUE
1B69 0100            CLR           YES. CLEAR ALL
1B6A 0089            SMB    ALLG   GREEN FLAG TO
1B6B 7479            STW    ALLG   TURN ENTIRE RAMP GREEN
1B6C 935D  RTALCR    LDX    ALLCLR-1
1B6D 2800            JSX  * 0   RETURN
```

```
                ' FILTER SUBPROGRAM  #24
1B6E 0000                D     0
1B6F 0040       FILTER   SLM
1B70 636E                STX   $-2    STORE INDEX FOR RETURN
1B71 0089                SMB   NBND
1B72 8400                LDW   NBND
1B73 73AA                STW   NBTEST
1B74 0800                SAZ        ANY BANDS IN TABLE?
1B75 1377                JMP   $+2
1B76 1395                JMP   RTFILTER   NO. RETURN
1B77 0100                CLR      YES. CLEAR
1B78 703B                STW   FPBC   PROCESSED BAND COUNTER
1B79 703D                STW   FTIX   TABLE INDEX
1B7A 903D       CETA     LDX   FTIX   TABLE INDEX
1B7B 0089                SMB   GBT
1B7C 8C07                LDW * GBT+6   ETA OF L EDGE OF THIS BAND
1B7D B04F                SUB   LHDY
1B7E B046                SUB   LHDYTLFL
1B7F 7042                STW   LTAMLHDY
1B80 0089                SMB   GBT
1B81 8C08                LDW * GBT+7   ETA OF T EDGE OF THIS BAND
1B82 A050                ADD   THDY
1B83 A047                ADD   THDYTLFL
1B84 7043                STW   TTAPTHDY
1B85 8049                LDW   RGPTOLFL
1B86 704A                STW   RGAPTOL
1B87 2304                JSX   LNDL   CHECK IF BAND
1B88 0040                SLM          SHOULD BE DELETED
1B89 803E                LDW   LDFG   DELETE BAND?
1B8A 0810                SAP
1B8B 1397                JMP   DELB   YES
1B8C 803D                LDW   FTIX   NO. BUMP TBL INDEX BY 8
1B8D A771                ADD   =8
1B8E 703D                STW   FTIX
1B8F 803B       ENDF     LDW   FPBC   INCREMENT
1B90 A75B                ADD   =1     PROCESSED BAND
1B91 703B                STW   FPBC   COUNTER BY 1
1B92 F3AA                CMW   NBTEST   PROCESSED ALL BANDS?
1B93 0860                SEQ
1B94 137A                JMP   CETA   NO. CALCULATE L AND T ETAS
1B95 936E       RTFILTER LDX   FILTER-1
1B96 2800                JSX * 0
```

```
          ' DELB SUBROUTINE - USED BY FILTER  #25
1B97 0089  DELB      SMB    NBND
1B98 8400            LDW    NBND
1B99 0A13            SLL    3
1B9A B03D            SUB    FTIX
1B9B 7056            STW    TMTM
1B9C 803D            LDW    FTIX    SET INDEX TO REF ENTRY AFTER
1B9D A771            ADD    =8      ENTRY TO BE DELETED
1B9E 0130            CAX
1B9F 0089  DELETE    SMB    GBT
1BA0 8C01            LDW  * GBT
1BA1 0089            SMB    GBT
1BA2 7BF9            STW  * GBT-8
1BA3 8056            LDW    TMTM
1BA4 B75B            SUB    =1
1BA5 7056            STW    TMTM
1BA6 0800            SAZ      FINISHED DELETING?
1BA7 13A9            JMP    $+2
1BA8 13AC            JMP    $+4   YES
1BA9 0401            IXS    1   NO. BUMP INDEX-WILL ALWAYS SKIP
1BAA 0000  NBTEST    D      0
1BAB 139F            JMP    DELETE   MOVE UP NEXT WORD
1BAC 0089            SMB    NBND   DECREMENT NBND BY 1
1BAD 8400            LDW    NBND
1BAE B75B            SUB    =1
1BAF E00C            AND    K51F    MASK BANDS OUT IF K51  ACTIVE
1BB0 0089            SMB    NBND
1BB1 7400            STW    NBND
1BB2 138F            JMP    ENDF
```

```
                ' ONOF ROUTINE  #26
1BB3 0000                  D       0
1BB4 0040       ONOF       SLM
1BB5 63B3                  STX     $-2   STORE INDEX FOR RETURN
1BB6 8029                  LDW     FAFG   FLASHING AMBER ON-OFF FLAG
1BB7 0800                  SAZ        ON?
1BB8 13C7                  JMP     ON   YES
1BB9 8028                  LDW     FAOF   NO. TIME LIGHT TURNED OFF
1BBA A75D                  ADD     =X'1F4'
1BBB 7056                  STW     TMTM
'BBC 0080                  SMB     TIME
 BBD 8021                  LDW     TIME   LIGHT OFF MORE
1BBE B056                  SUB     TMTM   THAN A SECOND?
1BBF 0810                  SAP
1BC0 13D4                  JMP     RTNF   NO. RETURN
1BC1 0080                  SMB     TIME   YES
1BC2 8021                  LDW     TIME   TIME LIGHT
1BC3 7027                  STW     FAON   TURNED ON
1BC4 875C                  LDW     =-1   SET FLASHING AMBER ON-OFF
1BC5 7029                  STW     FAFG   FLAG TO ON
1BC6 13D4                  JMP     RTNF   RETURN
1BC7 8027       ON         LDW     FAON   TIME LIGHT TURNED ON
1BC8 A75D                  ADD     =X'1F4'
1BC9 7056                  STW     TMTM
1BCA 0080                  SMB     TIME
1BCB 8021                  LDW     TIME   LIGHT ON
1BCC B056                  SUB     TMTM   MORE THAN A SECOND?
1BCD 0810                  SAP
1BCE 13D4                  JMP     RTNF   NO. RETURN
1BCF 0080                  SMB     TIME   YES
1BD0 8021                  LDW     TIME   TIME LIGHT
1BD1 7028                  STW     FAOF   TURNED OFF
1BD2 0100                  CLR        SET FLASHING AMBER ON-OFF
1BD3 7029                  STW     FAFG   FLAG TO ON
1BD4 93B3       RTNF       LDX     ONOF-1
1BD5 2800                  JSX * 0   RETURN
```

```
               ' BINK ROUTINE  #27
1BD6 0000                 D      0
1BD7 0040        BINK     SLM
1BD8 63D6                 STX    $-2    STORE INDEX FOR RETURN
1BD9 0100                 CLR
1BDA 0089                 SMB    NBND
1BDB 7400                 STW    NBND   CLEAR NUMBER OF BANDS IN TABLE
1BDC 705B                 STW    TX     CLEAR TABLE INDEX
1BDD 705D                 STW    VEH1   CLEAR FIRST VEHICLE FLAG
1BDE 7035                 STW    BIND   CLEAR BAND INDICATOR
1BDF 9770                 LDX    =-14   INIT IXR TO REF SENSOR F1
1BE0 6034        BNCK     STX    SIXR
1BE1 0082                 SMB    PNTW
1BE2 8BD6                 LDW *  PNTW+14
1BE3 702A                 STW    BEGP   BEGINNING POINTER
1BE4 0082                 SMB    PNTW
1BE5 8BD7                 LDW *  PNTW+15
1BE6 702B                 STW    ENDP   END POINTER
1BE7 F02A                 CMW    BEGP   THIS HIGHWAY LIST EMPTY?
1BE8 0870                 SNE
1BE9 14B1                 JMP    BEMY   YES. CHECK NEXT BIN
1BEA 805D                 LDW    VEH1   NO. FIRST VEHICLE ON HIGHWAY?
1BEB 0800                 SAZ
1BEC 1407                 JMP    BNXV   NO
1BED 2499                 JSX    STPM   YES. USE VEH PARAM AS POSSIBLE
1BEE 0040                 SLM                  LEAD VEH PARAMETERS
1BEF 875C                 LDW    =-1    SET FIRST VEHICLE INDICATOR
1BF0 705D                 STW    VEH1
1BF1 902A                 LDX    BEGP   BEGINNING POINTER
1BF2 0082                 SMB    PNTW
1BF3 9800                 LDX *  0
1BF4 0082                 SMB    PNTW
1BF5 8800                 LDW *  0     VEHICLE ETA. STORE AS
1BF6 7058                 STW    TETA   TRAILING ETA
1BF7 875C                 LDW    =-1   SET BAND INDICATOR TO INDICATE
1BF8 7035                 STW    BIND  FIRST BAND BEING PROCESSED
1BF9 8638                 LDW    UF844    TAMPA MERGE POINT****
1BFA 0089                 SMB    GBT
1BFB 7403                 STW    GBT+2
1BFC 8005                 LDW    VMRG
1BFD 7637                 STW    UFVNOWX  MAX VEL FOR 1ST BAND ***
1BFE 875C                 LDW    =-1    T
1BFF 7040                 STW    LTA    T
1C00 8039                 LDW    LETA    T
1C01 7041                 STW    TTA    T
1C02 2453                 JSX    GPOK   SET UP FIRST BAND
1C03 0040                 SLM
1C04 0100                 CLR           CLEAR BAND INDICATOR
1C05 7035                 STW    BIND
1C06 141E                 JMP    INBP   CHK GAP BETWEEN 1ST TWO VEH
```

```
           ' BINK ROUTINE - CONTINUED #28
1C07 902A   BNXV   LDX   BEGP   BEGINNING POINTER
1C08 0082          SMB   PNTW
1C09 9800          LDX * 0
1C0A 8039          LDW   LETA   PREVIOUS ETA
1C0B A04B          ADD   BINKTOL   TOLERANCE
1C0C 0082          SMB   PNTW
1C0D B800          SUB * 0   CURRENT ETA .25 SECONDS .GT.
1C0E 0820          SAM          PREVIOUS   ETA?
1C0F 141E          JMP   INBP  NO. IGNORE VEHICLE
           *
     0800   LAMJ   EQU   X'800'
1C10 902A          LDX   BEGP
1C11 0082          SMB   LAMJ
1C12 9800          LDX * LAMJ
1C13 8039          LDW   LETA
1C14 0082          SMB   LAMJ
1C15 B800          SUB * LAMJ
           *
1C16 0820          SAM
1C17 141E          JMP   INBP
1C18 0110          CMP
1C19 B769          SUB   =X'3E8'
1C1A 0820          SAM
1C1B 1425          JMP   GAPT     YES
1C1C 2499          JSX   STPM
1C1D 0040          SLM
1C1E 802A   INBP   LDW   BEGP
1C1F A75B          ADD   =1
1C20 702A          STW   BEGP
1C21 F02B          CMW   ENDP   THIS HIGHWAY LIST EXHAUSTED?
1C22 0870          SNE
1C23 14B1          JMP   BEMY   YES. CHECK NEXT BIN
1C24 1407          JMP   BNXV   NO. CHECK NEXT VEHICLE IN BIN
```

```
               ' BINK ROUTINE - GAP TEST SEGMENT  #29
1C25 A769  GAPT     ADD   =X'3E8'
1C26 B03A           SUB   LLEN
1C27 7054           STW   MGAP   MGAP= TRAILING ETA-LEADING ETA
1C28 902A           LDX   BEGP
1C29 0082           SMB   PNTW
1C2A 9800           LDX * 0
1C2B 0082           SMB   PNTW
1C2C 8802           LDW * 2
1C2D 705A           STW   TVEL   TRAILING ETA
1C2E 8057           LDW   SVEL
1C2F 7052           STW   LVEL
1C30 0082           SMB   PNTW
1C31 8800           LDW * 0
1C32 7058           STW   TETA   TRAILING TI
1C33 0082           SMB   PNTW
1C34 8801           LDW * 1
1C35 7059           STW   TTI   TRAILING VELOCITY (9/7 1.000)
1C36 8039           LDW   LETA
1C37 7037           STW   GLTA   LEADING ETA
1C38 803C           LDW   LTI
1C39 7038           STW   GLTI   LEADING TI
1C3A 24E2           JSX   GRAM   CALCULATE REQUIRED GAP
1C3B 0040           SLM
1C3C 8054           LDW   MGAP
1C3D B055           SUB   RGAP   GAP ADEQUATE ?
1C3E 0820           SAM
1C3F 1443           JMP   GPK    YES. SET UP BAND IN GBT
1C40 2499           JSX   STPM   NO.
1C41 0040           SLM
1C42 141E           JMP   INBP   CHECK NEXT VEH IN BIN
1C43 803A  GPK      LDW   LLEN   STORE LEADING VEH LENGTH
1C44 705F           STW   VLNGTH
1C45 8039           LDW   LETA
1C46 7040           STW   LTA
1C47 902A           LDX   BEGP
1C48 0082           SMB   PNTW
1C49 9800           LDX * 0
1C4A 0082           SMB   PNTW
1C4B 8800           LDW * 0
1C4C 7041           STW   TTA
1C4D 2453           JSX   GPOK   SET UP BAND IN TABLE
1C4E 0040           SLM
1C4F 2499           JSX   STPM
1C50 0040           SLM
1C51 141E           JMP   INBP   CHECK NEXT VEH IN BIN
```

```
                ' GPOK ROUTINE   #30
1C52 0000                D     0
1C53 0040       GPOK     SLM
1C54 6452                STX   $-2   STORE INDEX FOR RETURN
1C55 905B                LDX   TX   TABLE INDEX
1C56 8035                LDW   BIND
1C57 F75C                CMW   =-1   PROCESSING FIRST BAND?
1C58 0870                SNE
1C59 1467                JMP   FBND   YES
1C5A 804F                LDW   LHDY   NO. STORE LEADING HEADWAY FOR
1C5B 7051                STW   LTHDY  USE IN XLTE
1C5C 24AD                JSX   XLTE   CALCU'.ATE LEADING EDGE
1C5D 0040                SLM
1C5E 905B                LDX   TX   TABLE INDEX
1C5F 0089                SMB   GBT
1C60 7C03                STW * GBT+2   STORE LEADING EDGE IN TABLE
1C61 8636                LDW   UFVNOW
1C62 7637                STW   UFVNOWX
1C63 8035                LDW   BIND
1C64 F75B                CMW   =1   PROCESSING LAST BAND?
1C65 0870                SNE
1C66 1473                JMP   SVL   YES. XTE SET TO 0
1C67 8058       FBND     LDW   TETA   NO.
1C68 7039                STW   LETA   TRAILING ETA
1C69 0100                CLR     SET VEHICLE LENGTH TO ZERO
1C6A 705F                STW   VLNGTH
1C6B 8050                LDW   THDY   STORE TRAILING HEADWAY FOR
1C6C 0110                CMP            USE IN XLTE
1C6D 7051                STW   LTHDY
1C6E 24AD                JSX   XLTE   CALCULATE TRAILING EDGE
1C6F 0040                SLM
1C70 905B                LDX   TX
1C71 0089                SMB   GBT
1C72 7C05                STW * GBT+4   STORE TRAILING EDGE IN TABLE
1C73 2478       SVL      JSX   TVB
1C74 0040                SLM
1C75 9452                LDX   GPOK-1
1C76 2800                JSX * 0   RETURN TO CALLING POINT
```

```
                ' TVB ROUTINE   #31
1C77 0000                D      0
1C78 0040       TVB      SLM
1C79 6477                STX    $-2    STORE INDEX FOR RETURN
1C7A 905B                LDX    TX
1C7B 0080                SMB    TIME
1C7C 8021                LDW    TIME
1C7D 0089                SMB    GBT
1C7E 7C01                STW *  GBT    STORE TIME BAND CREATED
1C7F 8637                LDW    UFVNOWX
1C80 0089                SMB    GBT
1C81 7C02                STW *  GBT+1  STORE UFVNOW AS BAND VELOCITY
1C82 0089                SMB    NBND
1C83 8400                LDW    NBND
1C84 A75B                ADD    =1
1C85 E00C                AND    K51F   DELETE ALL BANDS IF K51 ON
1C86 0089                SMB    NBND
1C87 7400                STW    NBND   INCREMENT BAND COUNT BY 1
1C88 0100                CLR
1C89 0089                SMB    GBT
1C8A 7C04                STW *  GBT+3
1C8B 0089                SMB    GBT
1C8C 7C06                STW *  GBT+5
1C8D 8040                LDW    LTA
1C8E 0089                SMB    GBT
1C8F 7C07                STW *  GBT+6
1C90 8041                LDW    TTA
1C91 0089                SMB    GBT
1C92 7C08                STW *  GBT+7
1C93 805B                LDW    TX
1C94 A771                ADD    =8
1C95 705B                STW    TX     BUMP TABLE INDEX BY 8
1C96 9477                LDX    TVB-1
1C97 2800                JSX *  0      RETURN
```



```
                ' STPM ROUTINE  #32
1C98 0000                D      0
1C99 0040       STPM     SLM
1C9A 6498                STX    $-2
1C9B 902A                LDX    BEGP
1C9C 0082                SMB    PNTW
1C9D 9800                LDX *  0
1C9E 0082                SMB    PNTW
1C9F 8800                LDW *  0
1CA0 7039                STW    LETA   STORE ETA
1CA1 0082                SMB    PNTW
1CA2 8801                LDW *  1
1CA3 703C                STW    LT1    STORE T1
1CA4 0082                SMB    PNTW
1CA5 8802                LDW *  2
1CA6 7057                STW    SVEL   STORE VELOCITY
1CA7 0082                SMB    PNTW
1CA8 8803                LDW *  3
1CA9 703A                STW    LLEN   STORE VEH LENGTH IN SEC
1CAA 9498                LDX    STPM-1
1CAB 2800                JSX *  0      RETURN
```

```
               'XLTE ROUTINE AND END OF BINK CODE  #33
1CAC 0000                     D     0
1CAD 64AC      XLTE           STX   $-1
1CAE 163C                     JMP   UFMMMODE
1CAF 94AC      UFRET          LDX   XLTE-1
1CB0 2800                     JSX * 0   RETURN
1CB1 9034      BEMY           LDX   SIXR   CHECKED ALL 7 BINS?
1CB2 0402                     IXS   2
1CB3 13E0                     JMP   BNCK   NO.BUMP INDEX,CHECK NEXT BIN
1CB4 875B                     LDW   =1   YES. SET BAND INDICATOR TO SHOW
1CB5 7035                     STW   BIND   CREATING LAST BAND IN GBT
1CB6 905B                     LDX   TX
1CB7 876D                     LDW   =X'8000'
1CB8 0089                     SMB   GBT
1CB9 7C05                     STW * GBT+4   SET TRAILING EDGE TO X'8000'
1CBA 803A                     LDW   LLEN   STORE LENGTH OF LEADING VEH
1CBB 705F                     STW   VLNGTH  FOR USE IN XLTE
1CBC 8039                     LDW   LETA    T
1CBD 7040                     STW   LTA    T
1CBE 875C                     LDW   =-1    T
1CBF 7041                     STW   TTA    T
1CC0 2453                     JSX   GPOK   SET UP BAND IN TABLE
1CC1 0040                     SLM
1CC2 93D6                     LDX   BINK-1
1CC3 2800                     JSX * 0
```

```
               'DATA TABLES FOR GAP COMPUTATIONS  #34
               *
               *  FOLLOWING CONSTANT VALUES DEFINED
               *  IN RAMP VEH PROCESSING ROUTINE,
               *  VOL III, PAGES 9-2 AND 10-2.
               *
1CC4 0190      HOCM      D     X'190'   0.8 SEC (7/9)
1CC5 0000      H2PM      D     0
1CC6 0000      F1FM      D     0
1CC7 0000      F1LM      D     0
1CC8 0000      F2FM      D     0
1CC9 0000      F2LM      D     0
1CCA 0000      F30M      D     0
1CCB 0000      H1CM      D     0
1CCC 0000      H2CM      D     0
1CCD 0000      H1PM      D     0
1CCE 0000      HLPM      D     0
1CCF 0000      TF1M      D     0
1CD0 0000      TLTM      D     0
1CD1 0000      TRSM      D     0
1CD2 0000      SI1M      D     0
1CD3 0591      ARCM      D     X'591'   2.85 (7/9)
1CD4 01CE      BPCM      D     X'1CE'   .925 (7/9)
1CD5 2C1A      CBNM      D     X'2C1A'   88.2 FT/SEC (9/7)
1CD6 1180      CSTM      D     X'1180'   17.5 (8/8)
1CD7 2900      CTAM      D     X'2900'   41 FT/SEC (8/8)
1CD8 0CCD      HABM      D     X'CCD'   0.1 (1/15)
1CD9 075A      02CM      D     X'75A'   14.7 PSI (9/7)
1CDA 2800      PONM      D     X'2800'   0.9/73.5 (7/23)
1CDB 0FA0      TDCM      D     X'FA0'   .125/1.024 (1/15)
1CDC 0E40      TXCM      D     X'E40'   .057/1.024 (0/16)
1CDD 1B80      TXPM      D     X'1B80'   .11/1.024 (0/16)
1CDE 18DC      TP3M      D     X'18DC'   .777 (3/13)
1CDF 2800      TZPM      D     X'2800'   .16/1.024 (0/16)
1CE0 0148      XZNM      D     X'148'   .01 (1/15)
```

```
                ' GRAM ROUTINE PAGE 1 OF 3  #35
1CE1 0000                 D      0
1CE2 64E1       GRAM      STX    $-1       REQUIRED GAP SIZE PGM
1CE3 8005                 LDW    VMRG
1CE4 0917                 SLA    7
1CE5 7066                 STW    VGRAM
1CE6 8052                 LDW    LVEL    LEAD VEHICLE VELOCITY 9/7
1CE7 F05A                 CMW    TVEL    TRAILING VEHICLE VEL
1CE8 0890                 SLE
1CE9 1504                 JMP    GOAM    YES
1CEA 74C7                 STW    FILM    NO. LEAD VEHICLE VELOCITY
1CEB 805A                 LDW    TVEL
1CEC 74C6                 STW    FIFM    TRAILING VEHICLE VELOCITY
1CED 2562                 JSX    FICM  CALC H2 FOR TRAILING VEH
1CEE 74CC       GRRM      STW    H2CM    STORE H2
1CEF 8066                 LDW    VGRAM
1CF0 F052                 CMW    LVEL
1CF1 0860                 SEQ
1CF2 14F6                 JMP    GRYM    NO
1CF3 0100                 CLR       YES. CLEAR H1
1CF4 74CB                 STW    H1CM
1CF5 151F                 JMP    GRZM
1CF6 0880       GRYM      SGR       RAMP VEH GAINING ON LEAD VEH?
1CF7 14FE                 JMP    GRWM    NO. RAMP VEH FALLING AWAY
1CF8 74C6                 STW    FIFM    YES. RAMP VEHICLE VELOCITY
1CF9 8052                 LDW    LVEL    LEAD VEHICLE VELOCITY
1CFA 74C7                 STW    FILM    STORE LEAD VEHICLE VELOCITY
1CFB 2562                 JSX    FICM  CALCULATE TIME (H1) FOR RAMP
                *                          VEHICLE TO DECELERATE
                *                          TO LEAD VEHICLE'S VELOCITY
1CFC 74CB       GYYM      STW    H1CM    STORE H1
1CFD 151F                 JMP    GRZM
1CFE 74C8       GRWM      STW    F2FM    RAMP VEHICLE VELOCITY
1CFF 8052                 LDW    LVEL    LEAD VEHICLE VELOCITY
1D00 74C9                 STW    F2LM    STORE LEAD VEHICLE VELOCITY
1D01 2580                 JSX    F2CM  CALCULATE TIME (H1) FOR RAMP
                *                          VEHICLE TO ACCELERATE TO
                *                          LEAD VEHICLE'S VELOCITY
1D02 74CB       GWWM      STW    H1CM    STORE H1
1D03 151F                 JMP    GRZM
```

```
                ' GRAM ROUTINE PAGE 2 OF 3  #36
1D04 805A       GOAM    LDW   TVEL   TRAILING VEHICLE VELOCITY
1D05 F066               CMW   VGRAM
1D06 0890               SLE          GAINING ON RAMP VEHICLE?
1D07 1518               JMP   GNAM   YES
1D08 8066               LDW   VGRAM
1D09 F052               CMW   LVEL   LEAD VEH VEL. RAMP VEHICLE
1D0A 0880               SGR          GAINING ON LEAD VEHICLE?
1D0B 1514               JMP   GPAM   NO. GAP OPENING AT BOTH ENDS.
1D0C 74C6               STW   F1FM   YES. STORE RAMP VEH VELOCITY
1D0D 8052               LDW   LVEL
1D0E 74C7               STW   F1LM   STORE LEAD VEHICLE VELOCITY
1D0F 2562               JSX   F1CM  CALCULATE TIME (H1) FOR RAMP
                *               VEHICLE TO DECELERATE TO LEAD
                *               VEHICLE'S VELOCITY
1D10 74CB       FRRM    STW   H1CM   STORE H1
1D11 0100               CLR          CLEAR H2 SINCE NO
1D12 74CC               STW   H2CM   TRAILING HEADWAY NEEDED
1D13 151F               JMP   GRZM
1D14 0100       GPAM    CLR          CLEAR H1 AND H2 SINCE
1D15 74CB               STW   H1CM   GAP IS OPENING AT BOTH ENDS
1D16 74CC               STW   H2CM
1D17 151F               JMP   GRZM
1D18 74C9       GNAM    STW   F2LM   TRAILING VEHICLE VELOCITY
1D19 8066               LDW   VGRAM
1D1A 74C8               STW   F2FM   STORE RAMP VEHICLE VELOCITY
1D1B 2580               JSX   F2CM  CALCULATE TIME (H2) FOR RAMP
                *              VEHICLE TO ACCELERATE TO
                *                TRAILING VEHICLE'S VELOCITY
1D1C 74CC       GNNM    STW   H2CM   STORE H2
1D1D 0100               CLR          CLR H1 SINCE THE LEAD VEH IS
1D1E 74CB               STW   H1CM   PULLING AWAY FROM THE RAMP VEH
1D1F 8052       GRZM    LDW   LVEL
1D20 74CA               STW   F30M
1D21 254B               JSX   F3CM
```

```
              1478 ' GRAM ROUTINE PAGE 3 OF 3  #37
  1D22 0081   1479           SPMUL
S 1D23 2490
  1D24 1CDE   1480           D      TP3M
  1D25 0000   1481           D      0
  1D26 74D1   1482           STW    TRSM
  1D27 8037   1483           LDW    GLTA
  1D28 B038   1484           SUB    GLT1
  1D29 0081   1485           SPMUL
S 1D2A 2490
  1D2B 1CD1   1486           D      TRSM
  1D2C 0000   1487           D      0
  1D2D 0912   1488           SLA    2
  1D2E A04F   1489           ADD    LHDY
  1D2F 74CD   1490           STW    H1PM
  1D30 805A   1491           LDW    TVEL
  1D31 74CA   1492           STW    F30M
  1D32 254B   1493           JSX    F3CM
  1D33 0081   1494           SPMUL
S 1D34 2490
  1D35 1CDE   1495           D      TP3M
  1D36 0000   1496           D      0
  1D37 74D1   1497           STW    TRSM
  1D38 8058   1498           LDW    TETA
  1D39 B059   1499           SUB    TT1
  1D3A 0081   1500           SPMUL
S 1D3B 2490
  1D3C 1CD1   1501           D      TRSM
  1D3D 0000   1502           D      0
  1D3E 0912   1503           SLA    2
  1D3F A050   1504           ADD    THDY
  1D40 74C5   1505           STW    H2PM
  1D41 84CB   1506 DWRM      LDW    H1CM
  1D42 A4CD   1507           ADD    H1PM
  1D43 74CE   1508           STW    HLPM
  1D44 84CC   1509           LDW    H2CM
  1D45 A4C5   1510           ADD    H2PM
  1D46 A772   1511           ADD    =X'7D'   LENGTH(SEC) OF 15' RAMP VEH
  1D47 A4CE   1512           ADD    HLPM
  1D48 7055   1513           STW    RGAP
  1D49 94E1   1514           LDX    GRAM-1
  1D4A 2800   1515           JSX *  0   RETURN
              1516
```

```
        F3CM ROUTINE   #38                                                PAGE  40

                 1517 ' F3CM ROUTINE   #38
  1D4B 64D2      1518 F3CM      STX    SI1M
  1D4C 84CA      1519           LDW    F3OM
  1D4D F4D5      1520           CMW    CBNM
  1D4E 0880      1521           SGR
  1D4F 1553      1522           JMP    TATM
  1D50 84E0      1523           LDW    XZNM
  1D51 94D2      1524           LDX    SI1M
  1D52 2800      1525           JSX *  0
  1D53 F4D9      1526 TATM      CMW    O2CM
  1D54 0890      1527           SLE
  1D55 1559      1528           JMP    BKUM
  1D56 84D8      1529           LDW    HA3M
  1D57 94D2      1530           LDX    SI1M
  1D58 2800      1531           JSX *  0
  1D59 84D5      1532 BKUM      LDW    CBNM
  1D5A B4CA      1533           SUB    F3OM
  1D5B 0081      1534           SPMUL
S 1D5C 2490
  1D5D 1CDA      1535           D      PONM
  1D5E 0000      1536           D      0
  1D5F A4E0      1537           ADD    XZNM
  1D60 94D2      1538           LDX    SI1M
  1D61 2800      1539           JSX *  0
                 1540

        F1CM ROUTINE   #39                                                PAGE  41

                 1541 ' F1CM ROUTINE    #39
  1D62 64D2      1542 F1CM      STX    SI1M
  1D63 84C6      1543           LDW    F1FM   TRAILING VEHICLE VELOCITY
  1D64 B4C7      1544           SUB    F1LM   LEAD VEHICLE VELOCITY
  1D65 0911      1545           SLA    1
  1D66 08A0      1546           SNO
  1D67 1748      1547           JMP    UFPRA
  1D68 74CF      1548           STW    TF1M
  1D69 84C7      1549           LDW    F1LM
  1D6A 0911      1550           SLA    1
  1D6B 74C7      1551           STW    F1LM
  1D6C 84CF      1552           LDW    TF1M
  1D6D 0081      1553           SPMUL
S 1D6E 2490
  1D6F 1CDB      1554           D      TDCM
  1D70 0000      1555 LSHM      D      0
  1D71 F4C7      1556           CMW    F1LM
  1D72 0890      1557           SLE
  1D73 174D      1558           JMP    UFPRB
  1D74 0081      1559           SPDIV
S 1D75 24E0
  1D76 1D70      1560           D      LSHM
  1D77 1CC7      1561           D      F1LM
  1D78 0000      1562           D      0
  1D79 0081      1563           SPMUL
S 1D7A 2490
  1D7B 1CCF      1564           D      TF1M
  1D7C 0000      1565 LS2M      D      0
  1D7D 0911      1566           SLA    1
  1D7E 94D2      1567           LDX    SI1M
  1D7F 2800      1568           JSX *  0
                 1569
```

```
                ' F2CM ROUTINE  #40
  1D80 64D2     F2CM     STX    SIIM
  1D81 84C9              LDW    F2LM
  1D82 B4C8              SUB    F2FM
  1D83 0911              SLA    1
  1D84 08A0              SNO
  1D85 1752              JMP    UFPRC
  1D86 74D0              STW    TLTM
  1D87 F4D6              CMW    CSTM
  1D88 0890              SLE
  1D89 1590              JMP    FXCM
  1D8A 84D0              LDW    TLTM
  1D8B 0081              SPMUL
S 1D8C 2490
  1D8D 1CDC              D      TXCM
  1D8E 0000              D      0
  1D8F 159E              JMP    ATCM
  1D90 F4D7     FXCM     CMW    CTAM
  1D91 0890              SLE
  1D92 1599              JMP    FBCM
  1D93 0081              SPMUL
S 1D94 2490
  1D95 1CDD              D      TXPM
  1D96 0000              D      0
  1D97 B4D4              SUB    BPCM
  1D98 159E              JMP    ATCM
  1D99 0081     FBCM     SPMUL
S 1D9A 2490
  1D9B 1CDF              D      TZPM
  1D9C 0000              D      0
  1D9D B4D3              SUB    ARCM
  1D9E 94D2     ATCM     LDX    SIIM
  1D9F 2800              JSX *  0
  1DA0 0080     EXIM     SMB    JFRM
  1DA1 1020              JMP    JFRM
```

WINKOMATIC OVERHEAD SIGN PGM #41 PAGE 43

```
                'WINKOMATIC OVERHEAD SIGN  PGM  #41
  1DA2 0000              DATA   0
  1DA3 65A2     SSCK     STX    $-1    SAVE RETURN
  1DA4 8002              LDW    MODE
  1DA5 F75F              CMW    =3     MOVING MODE?
  1DA6 0870              SNE
  1DA7 15AB              JMP    SS02   YES. ACTIVATE DBGB MSG.
  1DA8 8773              LDW    =X'200'   NO. ACTIVATE PTS MSG.
  1DA9 7063     SS01     STW    SPSN
  1DAA 15B2              JMP    RTSS
                *  FOLLOWING CODE SETS OVHD SIGN
                *  TO PTS WHEN MM GREEN BANDS ARE
                *  MASKED.
  1DAB 800C     SS02     LDW    K51F
  1DAC 0810              SAP
  1DAD 15B0              JMP    SS03
  1DAE 8773              LDW    =X'200'   SET UP PTS MSG
  1DAF 15A9              JMP    SS01
  1DB0 8774     SS03     LDW    =X'400'   SET UP DBGB MSG
  1DB1 15A9              JMP    SS01
  1DB2 95A2     RTSS     LDX    SSCK-1
  1DB3 2800              JSX *  0   RETURN
```

```
                 ' YIELD SIGN CONTROL PGM  #42
1DB4 0000                  DATA   0
1DB5 0040        MWC       SLM
1DB6 65B4                  STX    $-2    STORE INDEX FOR RETURN
1DB7 800C                  LDW    K51F    IF GB IS MASKED (MERGE
1DB8 0820                  SAM            AREA OCCUPIED) TURN ON
1DB9 15D3                  JMP    CK02    YIELD SIGN
1DBA 8002                  LDW    MODE    ALSO TURN SIGN ON
1DBB F75B                  CMW    =1      IF IN INIT OR METERING
1DBC 0880                  SGR            MODE
1DBD 15D3                  JMP    CK02
1DBE 02F3                  DIN    X'F',3
1DBF E628                  AND    TSR3    TAMPA SENSOR R3 ACTIVATED?
1DC0 0800                  SAZ
1DC1 15C3                  JMP    $+2
1DC2 15C9                  JMP    CK015   NO
1DC3 0082        CK01      SMB    DBUF
1DC4 8031                  LDW    DBUF+8    YES IS THERE A GREEN
1DC5 E62B                  AND    TR3DM   BAND AT R3 NOW?
1DC6 0800                  SAZ
1DC7 15C9                  JMP    $+2
1DC8 15D3                  JMP    CK02
     1DC9        CK015     EQU    $
1DC9 02F5                  DIN    15,5
1DCA E62C                  AND    TSR4
1DCB 0800                  SAZ
1DCC 15CE                  JMP    $+2
1DCD 15DA                  JMP    CK03
1DCE 0082                  SMB    DBUF
1DCF 802F                  LDW    DBUF+6
1DD0 E62D                  AND    TR4DM
1DD1 0800                  SAZ
1DD2 15DA                  JMP    CK03
                 *-
1DD3 8775        CK02      LDW    =X'8002'    NO.  GO TURN ON MERGE
1DD4 7060                  STW    MWCS    WITH CAUTION SIGN
1DD5 0080                  SMB    TIME    AND SAVE TIME + 5 SECONDS
1DD6 8021                  LDW    TIME
1DD7 A062                  ADD    MWCONTM
1DD8 7061                  STW    MWON
1DD9 15E4                  JMP    RTMWC    RETURN
1DDA 8060        CK03      LDW    MWCS    IS SIGN ON
1DDB 0820                  SAM
1DDC 15E4                  JMP    RTMWC    RETURN
1DDD 0080                  SMB    TIME    YES
1DDE 8021                  LDW    TIME    TIMER RUN OUT?
1DDF B061                  SUB    MWON
1DE0 0810                  SAP
1DE1 15E4                  JMP    RTMWC    RETURN
1DE2 0100                  CLR     YES
1DE3 7060                  STW    MWCS
1DE4 95B4        RTMWC     LDX    MWC-1
1DE5 2800                  JSX * 0   RETURN
```

```
              1680 ' RAMP QUEUEING PGM   #43
  1DE6 0000
  1DE7 65E6   1681 CHKK19    SUBR
  1DE8 0080   1682           SMB    RLTQUE  GET QUEUE STATUS
  1DE9 8048   1683           LDW    RLTQUE
  1DEA 0820   1684           SAM        IF NEGATIVE THEN CLEAR
  1DEB 15EE   1685           JMP    TIM1
  1DEC 802E   1686           LDW    K19NOQSG  SO USE LONG RED TIME
  1DED 15EF   1687           JMP    K19OUT
  1DEE 802D   1688 TIM1      LDW    K19QSG   IF QUEUED THEN SHORT
  1DEF 701B   1689 K19OUT    STW    MAXRED   RED TIME
              1690 *-
  1DF0 02F5   1691           DIN    15,5    R5-R1 CHECK
  1DF1 E776   1692           AND    =2496
  1DF2 7631   1693           STW    WRK
  1DF3 02F3   1694           DIN    15,3
  1DF4 E777   1695           AND    =4
  1DF5 C631   1696           ORI    WRK
  1DF6 E62F   1697           AND    RMPS
  1DF7 762F   1698           STW    RMPS
  1DF8 0800   1699           SAZ
  1DF9 1601   1700           JMP    RMPCT
  1DFA 7604   1701           STW    RCNT
  1DFB 875C   1702           LDW    =-1
  1DFC 762F   1703           STW    RMPS
  1DFD 0080   1704           SMB    TIME
  1DFE 8021   1705           LDW    TIME
  1DFF 7603   1706           STW    RCLK
  1E00 1607   1707           JMP    MERCHK
       1E01   1708 RMPCT     EQU    $
  1E01 008F   1709           CUSC
S 1E02 26E8
  1E03 0000   1710 RCLK      D      0
  1E04 0000   1711 RCNT      D      0
  1E05 0004   1712 RLIM      D      4  SEC
  1E06 161C   1713           JMP    QHMAX
       1E07   1714 MERCHK    EQU    $
  1E07 02F3   1715           DIN    15,3    M1, M2 CHECK
  1E08 E778   1716           AND    =2176
  1E09 E630   1717           AND    MAS
  1E0A 7630   1718           STW    MAS
  1E0B 0800   1719           SAZ
  1E0C 1614   1720           JMP    MACNT
  1E0D 7617   1721           STW    MCNT
  1E0E 875C   1722           LDW    =-1
  1E0F 7630   1723           STW    MAS
  1E10 0080   1724           SMB    TIME
  1E11 8021   1725           LDW    TIME
  1E12 7616   1726           STW    MCLK
  1E13 161A   1727           JMP    CKKEXT
       1E14   1728 MACNT     EQU    $
  1E14 008F   1729           CUSC
S 1E15 26E8
  1E16 0000   1730 MCLK      D      0
  1E17 0000   1731 MCNT      D      0
  1E18 000A   1732 MLIM      D      10  SEC
  1E19 161C   1733           JMP    QHMAX
```

```
      1E1A  1734 CKKEXT   EQU   $
1E1A 95E6   1735          EXIT  CHKK19
1E1B 2800
      1E1C  1736 QHMAX    EQU  $
1E1C 806A   1737          LDW   MAXREDQ
1E1D 701B   1738          STW   MAXRED
1E1E 0080   1739          SMB   TIME   KEEP TIME OVERFLOWING
1E1F 8021   1740          LDW   TIME
1E20 B75D   1741          SUB   =500
1E21 7616   1742          STW   MCLK   FOR MERGE AREA CLOCK
1E22 7603   1743          STW   RCLK    AND RAMP CLOCK
1E23 8605   1744          LDW   RLIM   SECOND COUNTS FOR
1E24 7604   1745          STW   RCNT    RAMP
1E25 8618   1746          LDW   MLIM   AND MERGE AREA
1E26 7617   1747          STW   MCNT
1E27 161A   1748          JMP   CKKEXT
            1749
```



```
            1750 ' UF SENSOR MASKS  #44
            1751 *
            1752 *  TAMPA RAMP SENSOR MASKS
            1753 *
1E28 0004   1754 TSR3     D     X'0004'
1E29 0200   1755 TSR6A    D     X'0200'
1E2A 0400   1756 TSR6B    D     X'0400'
1E2B F000   1757 TR3DM    D     X'F000'  MASK FOR GB LIGHTS
            1758 *                        OPPOSITE R3
1E2C 0080   1759 TSR4     D     X'80'   R4 SENSOR
1E2D 0007   1760 TR4DM    D     7    LIGHTS OPPOSITE R4
1E2E 0020   1761 TSR7     D     X'20'     R7 SENSOR
1E2F        1762 RMPS     RES   1
1E30        1763 MAS      RES   1
1E31        1764 WRK      RES   1
            1765
```

```
                ' UF ROUTINE   #45
                *
                * VARIABLE SPEED GREEN BANDS
                *
      20CC      UFPAM     EQU    X'20CC'
      20CD      UFPLA     EQU    X'20CD'
      20CE      UFPBL     EQU    X'20CE'
  1E32 0000     UFTMP     DATA   0
  1E33 0000     UFTAE     DATA   0
  1E34 0000     UFTSL     DATA   0
  1E35 0000     UFTDB     DATA   0
  1E36 0000     UFVNOW    DATA   0
  1E37 0000     UFVNOWX   DATA   0
  1E38 034C     UF844     DATA   844
  1E39 0625     UF1573    DATA   1573
  1E3A 0164     UF356     DATA   356
  1E3B 0B44     UF2884    DATA   2884
                *
  1E3C 8039     UFMMMODE LDW     LETA
  1E3D A051               ADD    LTHDY
  1E3E A05F               ADD    VLNGTH
  1E3F 0080               SMB    TIME
  1E40 B021               SUB    TIME
  1E41 7632               STW    UFTMP    TIME TO MERGE POINT
  1E42 0088               SMB    UFPAM
  1E43 B0CC               SUB    UFPAM
  1E44 7633               STW    UFTAE
  1E45 0088               SMB    UFPLA
  1E46 B0CD               SUB    UFPLA
  1E47 7634               STW    UFTSL
  1E48 0088               SMB    UFPBL
  1E49 B0CE               SUB    UFPBL
  1E4A 7635               STW    UFTDB
  1E4B 8632               LDW    UFTMP
  1E4C 0810               SAP
  1E4D 1655               JMP    UFATMP
  1E4E 8633               LDW    UFTAE
  1E4F 0810               SAP
  1E50 165A               JMP    UFFAST
  1E51 8634               LDW    UFTSL
  1E52 0810               SAP
  1E53 1668               JMP    UFACC
  1E54 1676               JMP    UFSLOW
  1E55 8005     UFATMP    LDW    VMRG
  1E56 7636               STW    UFVNOW
  1E57 8638               LDW    UF844
  1E58 0040               SLM
  1E59 14AF               JMP    UFRET
  1E5A 8005     UFFAST    LDW    VMRG
  1E5B 7636               STW    UFVNOW
  1E5C B757               SUB    =44
  1E5D 0130               CAX
  1E5E 0040               SLM
  1E5F 8E80               LDW *  FASTBL
  1E60 0081               SPMUL
S 1E61 2490
  1E62 1E32               DATA   UFTMP
```

```
  1E63 0000  1822          DATA   0
  1E64 0110  1823          CMP
  1E65 A638  1824          ADD    UF844
  1E66 0040  1825          SLM
  1E67 14AF  1826          JMP    UFRET
  1E68 8639  1827 UFACC    LDW    UF1573
  1E69 0081  1828          SPMUL
S 1E6A 2490
  1E6B 1E34  1829          DATA   UFTSL
  1E6C 0000  1830          DATA   0
  1E6D 0110  1831          CMP
  1E6E 0130  1832          CAX
  1E6F A63A  1833          ADD    UF356
  1E70 0903  1834          SRA    3
  1E71 C76D  1835          ORI    =X'8000'
  1E72 7636  1836          STW    UFVNOW
  1E73 0040  1837          SLM
  1E74 8E97  1838          LDW *  UFEDGE
  1E75 14AF  1839          JMP    UFRET
  1E76 8757  1840 UFSLOW   LDW    =44
  1E77 7636  1841          STW    UFVNOW
  1E78 8635  1842          LDW    UFTDB
  1E79 0081  1843          SPMUL
S 1E7A 2490
  1E7B 1E3B  1844          DATA   UF2884
  1E7C 0000  1845          DATA   0
  1E7D 0040  1846          SLM
  1E7E 0110  1847          CMP
  1E7F 14AF  1848          JMP    UFRET
             1849
```



```
             1850 '  DATA TABLES FOR UF MM LOGIC    #46
  1E80 0B44  1851 FASTBL   DATA   2884
  1E81 0B35  1852          DATA   2949
  1E82 0BC7  1853          DATA   3015
  1E83 0C08  1854          DATA   3080
  1E84 0C4A  1855          DATA   3146
  1E85 0C8B  1856          DATA   3211
  1E86 0CCD  1857          DATA   3277
  1E87 0D0E  1858          DATA   3342
  1E88 0D50  1859          DATA   3408
  1E89 0D91  1860          DATA   3473
  1E8A 0DD3  1861          DATA   3539
  1E8B 0E14  1862          DATA   3604
  1E8C 0E56  1863          DATA   3670
  1E8D 0E98  1864          DATA   3736
  1E8E 0ED9  1865          DATA   3801
  1E8F 0F1B  1866          DATA   3867
  1E90 0F5C  1867          DATA   3932
  1E91 0F9E  1868          DATA   3998
  1E92 0FDF  1869          DATA   4063
  1E93 1021  1870          DATA   4129
  1E94 1062  1871          DATA   4194
  1E95 10A4  1872          DATA   4260
  1E96 10E5  1873          DATA   4325
             1874
             1875
             1876
```

```
               'DATA TABLES FOR UF MM MODE    #47
1E97 0108  UFEDGE   DATA  264
1E98 010A           DATA  266
1E99 010B           DATA  267
1E9A 010D           DATA  269
1E9B 010F           DATA  271
1E9C 0111           DATA  273
1E9D 0113           DATA  275
1E9E 0115           DATA  277
1E9F 0117           DATA  279
1EA0 0119           DATA  281
1EA1 011B           DATA  283
1EA2 011C           DATA  284
1EA3 011E           DATA  286
1EA4 0120           DATA  288
1EA5 0122           DATA  290
1EA6 0124           DATA  292
1EA7 0126           DATA  294
1EA8 0128           DATA  296
1EA9 012A           DATA  298
1EAA 012C           DATA  300
1EAB 012E           DATA  302
1EAC 0130           DATA  304
1EAD 0132           DATA  306
1EAE 0134           DATA  308
1EAF 0136           DATA  310
1EB0 0137           DATA  311
1EB1 0139           DATA  313
1EB2 013B           DATA  315
1EB3 013D           DATA  317
1EB4 013F           DATA  319
1EB5 0141           DATA  321
1EB6 0143           DATA  323
1EB7 0145           DATA  325
1EB8 0147           DATA  327
1EB9 0149           DATA  329
1EBA 014B           DATA  331
1EBB 014D           DATA  333
1EBC 014F           DATA  335
1EBD 0151           DATA  337
1EBE 0153           DATA  339
1EBF 0156           DATA  342
1EC0 0158           DATA  344
1EC1 015A           DATA  346
1EC2 015C           DATA  348
1EC3 015E           DATA  350
1EC4 0160           DATA  352
1EC5 0162           DATA  354
1EC6 0164           DATA  356
1EC7 0166           DATA  358
1EC8 0168           DATA  360
1EC9 016A           DATA  362
1ECA 016C           DATA  364
1ECB 016E           DATA  366
1ECC 0170           DATA  368
1ECD 0173           DATA  371
1ECE 0175           DATA  373
```

1ECF	0177	1934	DATA	375
1ED0	0179	1935	DATA	377
1ED1	017B	1936	DATA	379
1ED2	017D	1937	DATA	381
1ED3	017F	1938	DATA	383
1ED4	0182	1939	DATA	386
1ED5	0184	1940	DATA	388
1ED6	0186	1941	DATA	390
1ED7	0188	1942	DATA	392
1ED8	018A	1943	DATA	394
1ED9	018C	1944	DATA	396
1EDA	018F	1945	DATA	399
1EDB	0191	1946	DATA	401
1EDC	0193	1947	DATA	403
1EDD	0195	1948	DATA	405
1EDE	0197	1949	DATA	407
1EDF	019A	1950	DATA	410
1EE0	019C	1951	DATA	412
1EE1	019E	1952	DATA	414
1EE2	01A0	1953	DATA	416
1EE3	01A2	1954	DATA	418
1EE4	01A5	1955	DATA	421
1EE5	01A7	1956	DATA	423
1EE6	01A9	1957	DATA	425
1EE7	01AB	1958	DATA	427
1EE8	01AE	1959	DATA	430
1EE9	01B0	1960	DATA	432
1EEA	01B2	1961	DATA	434
1EEB	01B4	1962	DATA	436
1EEC	01B7	1963	DATA	439
1EED	01B9	1964	DATA	441
1EEE	01BB	1965	DATA	443
1EEF	01BE	1966	DATA	446
1EF0	01C0	1967	DATA	448
1EF1	01C2	1968	DATA	450
1EF2	01C4	1969	DATA	452
1EF3	01C7	1970	DATA	455
1EF4	01C9	1971	DATA	457
1EF5	01CB	1972	DATA	459
1EF6	01CE	1973	DATA	462
1EF7	01D0	1974	DATA	464
1EF8	01D2	1975	DATA	466
1EF9	01D5	1976	DATA	469
1EFA	01D7	1977	DATA	471
1EFB	01D9	1978	DATA	473
1EFC	01DC	1979	DATA	476
1EFD	01DE	1980	DATA	478
1EFE	01E0	1981	DATA	480
1EFF	01E3	1982	DATA	483
1F00	01E5	1983	DATA	485
1F01	01E8	1984	DATA	488
1F02	01EA	1985	DATA	490
1F03	01EC	1986	DATA	492
1F04	01EF	1987	DATA	495
1F05	01F1	1988	DATA	497
1F06	01F4	1989	DATA	500
1F07	01F6	1990	DATA	502

1F08	01F8	1991	DATA	504
1F09	01FB	1992	DATA	507
1F0A	01FD	1993	DATA	509
1F0B	0200	1994	DATA	512
1F0C	0202	1995	DATA	514
1F0D	0205	1996	DATA	517
1F0E	0207	1997	DATA	519
1F0F	020A	1998	DATA	522
1F10	020C	1999	DATA	524
1F11	020E	2000	DATA	526
1F12	0211	2001	DATA	529
1F13	0213	2002	DATA	531
1F14	0216	2003	DATA	534
1F15	0218	2004	DATA	536
1F16	021B	2005	DATA	539
1F17	021D	2006	DATA	541
1F18	0220	2007	DATA	544
1F19	0222	2008	DATA	546
1F1A	0225	2009	DATA	549
1F1B	0227	2010	DATA	551
1F1C	022A	2011	DATA	554
1F1D	022C	2012	DATA	556
1F1E	022F	2013	DATA	559
1F1F	0232	2014	DATA	562
1F20	0234	2015	DATA	564
1F21	0237	2016	DATA	567
1F22	0239	2017	DATA	569
1F23	023C	2018	DATA	572
1F24	023E	2019	DATA	574
1F25	0241	2020	DATA	577
1F26	0243	2021	DATA	579
1F27	0246	2022	DATA	582
1F28	0249	2023	DATA	585
1F29	024B	2024	DATA	587
1F2A	024E	2025	DATA	590
1F2B	0250	2026	DATA	592
1F2C	0253	2027	DATA	595
1F2D	0256	2028	DATA	598
1F2E	0258	2029	DATA	600
1F2F	025B	2030	DATA	603
1F30	025D	2031	DATA	605
1F31	0260	2032	DATA	608
1F32	0263	2033	DATA	611
1F33	0265	2034	DATA	613
1F34	0268	2035	DATA	616
1F35	026B	2036	DATA	619
1F36	026D	2037	DATA	621
1F37	0270	2038	DATA	624
1F38	0273	2039	DATA	627
1F39	0275	2040	DATA	629
1F3A	0278	2041	DATA	632
1F3B	027B	2042	DATA	635
1F3C	027D	2043	DATA	637
1F3D	0280	2044	DATA	640
1F3E	0283	2045	DATA	643
1F3F	0286	2046	DATA	646
1F40	0288	2047	DATA	648

```
1F41 028B  2048           DATA  651
1F42 028E  2049           DATA  654
1F43 0290  2050           DATA  656
1F44 0293  2051           DATA  659
1F45 0296  2052           DATA  662
1F46 0299  2053           DATA  665
1F47 029B  2054           DATA  667
           2055
```

```
              2056 'UF CODE TO PRINT INIT CONDITION #48
              2057 *
         0046 2058 ERRCH      EQU   X'046'   CORE LOC OF 'CAUSE' CHAR
              2059 *
1F48 0100     2060 UFPRA      CLR
1F49 0SC1     2061            LLB   X'C1'
1F4A 0080     2062            SMB   ERRCH
1F4B 7046     2063            STW   ERRCH
1F4C 15A0     2064            JMP   EXIM
1F4D 0100     2065 UFPRB      CLR
1F4E 0SC2     2066            LLB   X'C2'
1F4F 0080     2067            SMB   ERRCH
1F50 7046     2068            STW   ERRCH
1F51 15A0     2069            JMP   EXIM
1F52 0100     2070 UFPRC      CLR
1F53 0SC3     2071            LLB   X'C3'
1F54 0080     2072            SMB   ERRCH
1F55 7046     2073            STW   ERRCH
1F56 15A0     2074            JMP   EXIM
              2075 *
              2076            END

1F57 002C
1F58 003B
1F59 FFF6
1F5A FFFB
1F5B 0001
1F5C FFFF
1F5D 01F4
1F5E 17D4
1F5F 0005
1F60 0029
1F61 0038
1F62 1D4C
1F63 0020
1F64 0010
1F65 FFFE
1F66 0002
1F67 0025
1F68 0040
1F69 03E8
1F6A 1F40
1F6B 0033
1F6C 06D6
1F6D 8000
1F6E 0114
1F6F 00F4
1F70 FFF2
1F71 0008
1F72 007D
1F73 0200
1F74 0400
1F75 8002
1F76 09C0
1F77 0004
1F78 0880

NO ERRORS
```

ABNK	1B61	ACTF	1809	ALLCLR	1B5E	ALLG	2479
ARCM	1CD3	ARTI	1812	ATCM	1D9E	BEGP	182A
BEM	1B59	BEMY	1CB1	BGBS	1950	BIND	1835
BINK	1BD7	BINKTOL	184B	BIXR	2001	BKUM	1D59
BNCK	1BE0	BNK	1B0D	BNXV	1C07	BPCM	1CD4
CBNM	1CD5	CETA	1B7A	CHECKGRN	1AE9	CHKK19	1DE7
CIXR	2002	CK01	1DC3	CK015	1DC9	CK02	1D03
CK03	1DDA	CKDEMAND	1A74	CKKEXT	1E1A	CSTM	1CD6
CTAM	1CD7	CTMM	1A02	CUSC	3EE8	D	1836
DBUF	0829	DELB	1B97	DELETE	1B9F	DSTA	10BF
DWRM	1D41	ENDF	1B8F	ENDP	182B	ERRCH	0046
EXEC	0023	EXIM	1DA0	EXPFLG	1813	F1CM	1D62
F1FM	1CC6	F1LM	1CC7	F2CM	1D80	F2FM	1CC8
F2LM	1CC9	F3CM	1D4B	F30M	1CCA	FAFG	1829
FA0F	1828	FA0N	1827	FASTBL	1E80	FBCM	1D99
FBND	1C67	FILT	1A3A	FILTER	1B6F	FINT	1972
FLT	1A37	FOUR	1831	FPBC	183B	FRRM	1D10
FTIX	183D	FXCM	1D90	GAPT	1C25	GBS	18F1
GBSDONE	248F	GBSSTRT	247C	GBT	2401	GLT1	1838
GLTA	1837	GNAM	1D18	GNNM	1D1C	GOAM	1D04
GPAM	1D14	GPK	1C43	GPOK	1C53	GRAM	1CE2
GRNOPTFL	1814	GRNOPTST	197D	GRRM	1CEE	GRWM	1CFE
GRYM	1CF6	GRZM	1D1F	GWWM	1D02	GYYM	1CFC
H1CM	1CCB	H1PM	1CCD	H2CM	1CCC	H2PM	1CC5
HABM	1CD8	HISP	1864	HLPM	1CCE	HOCM	1CC4
IBP	1B53	INBP	1C1E	INFG	1804	INI	18AD
INONOF	1968	INTON	196F	INTREDFL	1824	INTTIM	1811
JFRM	0020	JMPMODE	197F	JVELVL	2000	K19NOQSG	182E
K19NOQSM	1830	K19OUT	1DEF	K19QSG	182D	K19QSM	182F
K19T	182C	K30F	1807	K30V	1808	K51F	180C
LAMJ	0800	LDFG	183E	LETA	1839	LHDY	184F
LHDYTLFL	1846	LHDYTLLN	1844	LIGHT	1815	LLEN	183A
LNDL	1B04	LS2M	1D7C	LSH	1867	LSHM	1D70
LT1	183C	LT4	1840	LTAMLHDY	1842	LTHDY	1851
LVEHFLAG	184C	LVEL	1852	MACNT	1E14	MAS	1E30
MAXAMBR	181A	MAXGREEN	1819	MAXRED	181B	MAXREDQ	186A
MCLK	1E16	MCNT	1E17	MDSP	1865	MERCHK	1E07
MGAP	1854	MINCLK	180A	MINRED	181C	MLIM	1E18
MM	198A	MMMODE	1A0E	MMSG	1A1C	MMSGAM	1821
MMSGFG	1820	MODE	1802	MODESG	1A9A	MODESGBF	1AA2
MTAB	18A8	MWC	1DB5	MWCONTM	1862	MWCS	1860
MWON	1861	NBND	2400	NBTEST	1BAA	NOKF	1B4B
NOKL	1ADF	NUMV	2004	NVEH	1B17	O2CM	1CD9
OLDS	1833	ON	1BC7	ONOF	1BB4	OPTFLG	1AFE
OUTPUT	1992	P500	1868	PATCH1A	1999	PATCH1B	199B
PNTW	0BC8	PONM	1CDA	PSF	180D	Q	1803
QHMAX	1E1C	QMMSG	184D	QSGMM	183F	R7CHK	191B
R7TEXT	193C	R7TOG	1869	R7TOGP	192D	RCLK	1E03
RCLR	186B	RCNT	1E04	REDFLAG	1853	REINIT	18C5
RGAP	1855	RGAPTOL	184A	RGPTOLFL	1849	RGPTOLLN	1848
RLIM	1E05	RLNBND	0836	RLTQUE	0048	RMPCT	1E01
RMPL	184E	RMPS	1E2F	RSCLR	1903	RTALCR	1B6C
RTFILT	1A41	RTFILTER	1B95	RTLD	1B4D	RTMWC	1DE4
RTNF	1BD4	RTSS	1DB2	SBUF	10C8	SCAST	1914
SCLK	1905	SCNT	1906	SCNT0	1909	SCOVR	190C
SEC4	1832	SETAMBER	1AF8	SETGREEN	1A81	SETMM	19FB
SETRED	1A4D	SETSM	199C	SG	1A07	SGMINRED	181D

SGMMBD	180F	SGMMFG	1823	SGMMT	1822	SGMODE	19BA
SIIM	1CD2	SIXR	1834	SLIM	1907	SM	19B1
SMANDSG	1A43	SMMODE	19A6	SMSGBD	180E	SPAR	1B4F
SPDIV	04E0	SPMUL	0490	SPSN	1863	SS01	1DA9
SS02	1DAB	SS03	1DB0	SSCK	1DA3	SSFLAG	1810
STATERED	1A63	STPM	1C99	STSG	1A29	SVEL	1857
SVL	1C73	TATM	1D53	TDCM	1CDB	TETA	1858
TFIM	1CCF	THDY	1850	THDYTLFL	1847	THDYTLLN	1845
THRUBFR	1A53	TIMI	1DEE	TIME	0021	TIMEAMBR	1817
TIMEGRN	1816	TIMERED	1818	TLTM	1CD0	TMTM	1856
TNMSCK	180B	TP3M	1CDE	TR3DM	1E2B	TR4DM	1E2D
TRSM	1CD1	TSGMM	19C9	TSR3	1E28	TSR4	1E2C
TSR6A	1E29	TSR6B	1E2A	TSR7	1E2E	TTI	1859
TTA	1841	TTAPTHDY	1843	TVB	1C78	TVEL	185A
TVL	2003	TWRED	19F0	TX	185B	TXCM	1CDC
TXPM	1CDD	TZPM	1CDF	UF1573	1E39	UF2884	1E3B
UF356	1E3A	UF844	1E38	UFACC	1E68	UFATMP	1E55
UFEDGE	1E97	UFFAST	1E5A	UFMMMODE	1E3C	UFPAM	20CC
UFPBL	20CE	UFPLA	20CD	UFPRA	1F48	UFPRB	1F4D
UFPRC	1F52	UFRET	1CAF	UFSLOW	1E76	UFTAE	1E33
UFTDB	1E35	UFTMP	1E32	UFTSL	1E34	UFVNOW	1E36
UFVNOWX	1E37	VBAR	1806	VBVL	185C	VEHI	185D
VGRAM	1866	VLNGTH	185F	VMRG	1805	VMRGTBL	186C
VWAITFLG	1825	VWAITIME	1826	VWAITING	1A7E	VWCLK	192F
VWCNT	1930	VWLIM	1931	WREDFLAG	181E	WREDTIME	181F
WRK	1E31	XLTE	1CAD	XZNM	1CE0	ZERO	185E

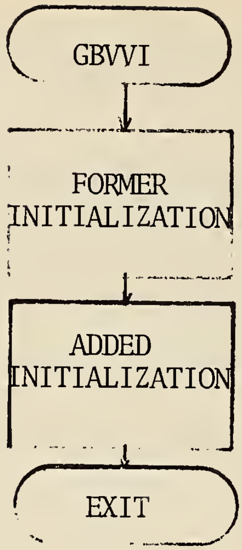
PAS?

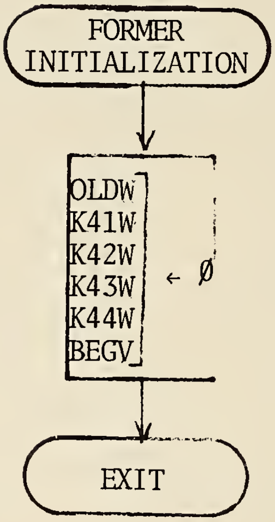
VELOCITY AND VOLUME SUBPROGRAM

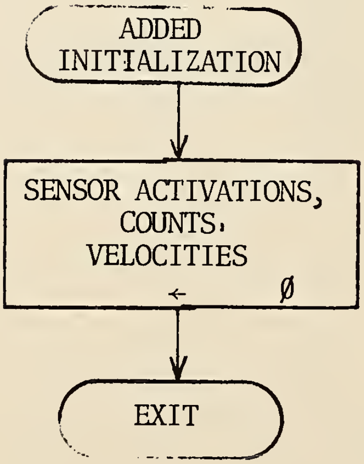
The basis for velocity determinations and volume computations were distributed from single sensors to groups. For velocity, sensors F1, F2, F3 are used and the "best" two of three are used for the final velocity determination. A ringbuffer stores 64 velocities for each sensor and keeps a running average. The ringbuffer is initialized by setting the current entry to Ø. When the value is discovered, the ringbuffer program uses the updated velocity that has been passed to initialize the entire ring and set a velocity value. The normal update is to pass a velocity to the ringbuffer program which replaces the oldest entry in the ring and adjusts the velocity average accordingly.

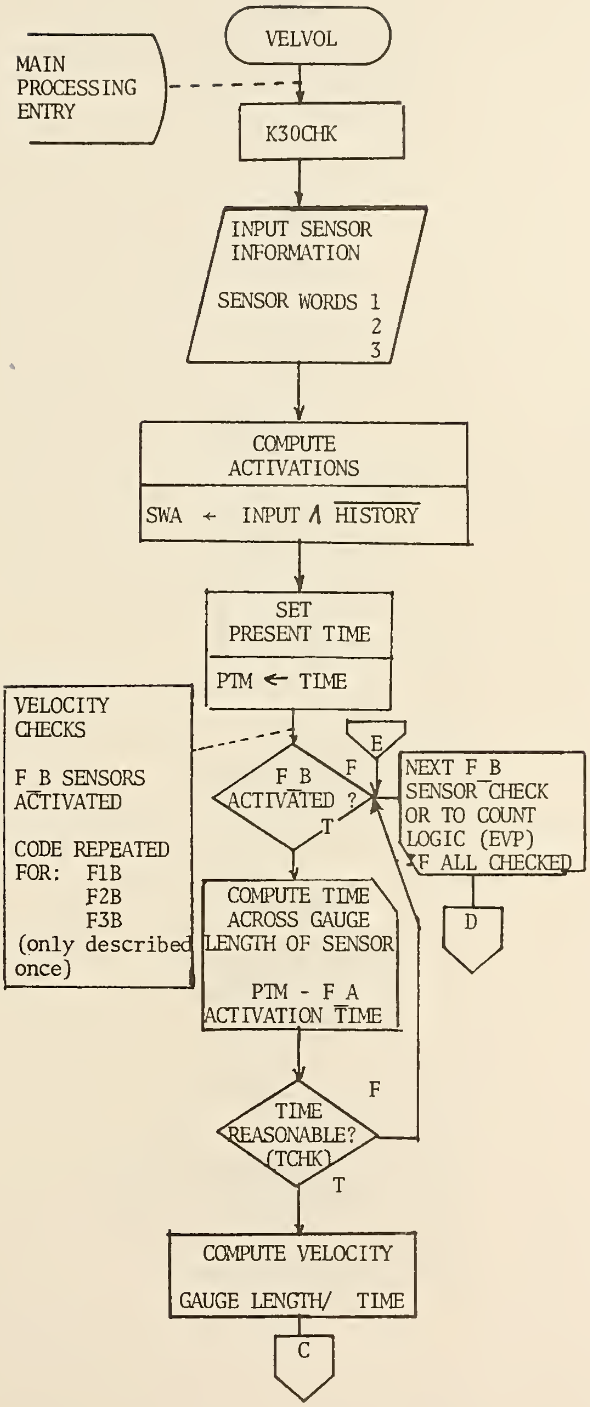
The volume determination uses a 3 minute sum on F1A, F2A, F3A, R1, R2, R3, then selects the "best" two of three from the F1A, F2A, F3A group and the "best" two of three from the R1, R2, R3 group. The two volumes are averaged and become the "last" 3-minute volume.

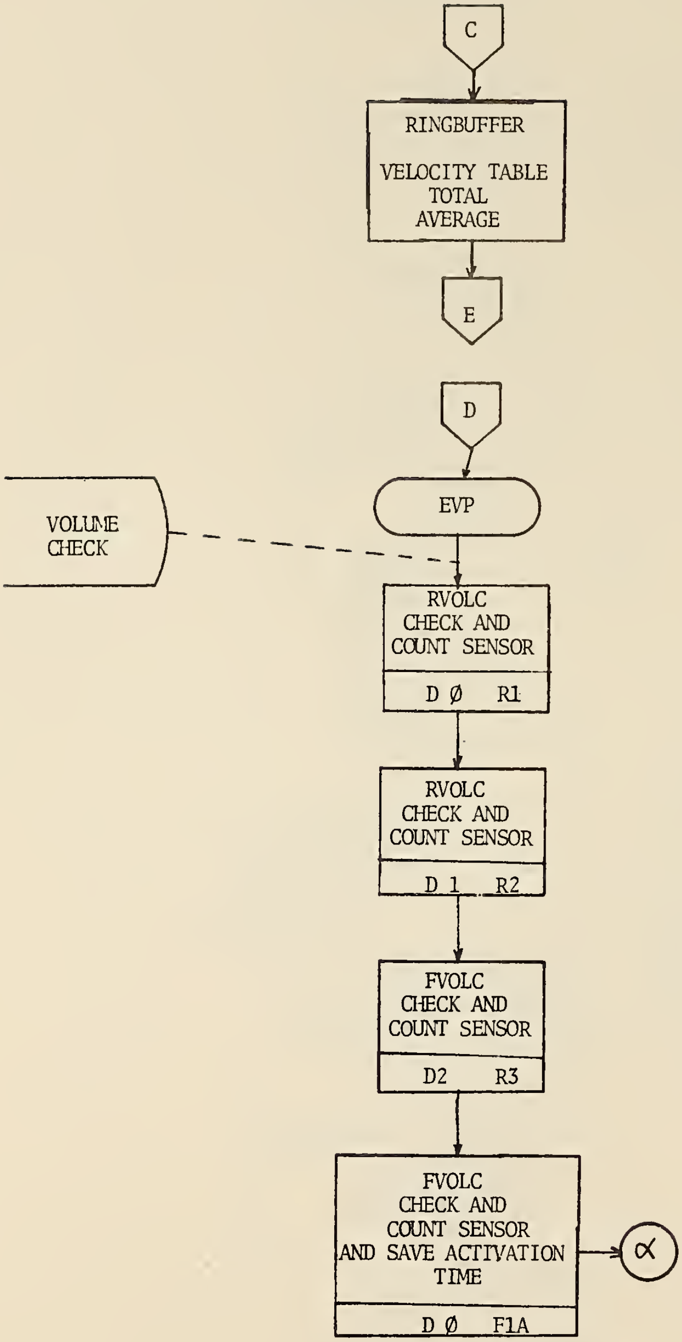
Highway Vehicle Subprogram no longer is used to pass information to Velocity and Volume.

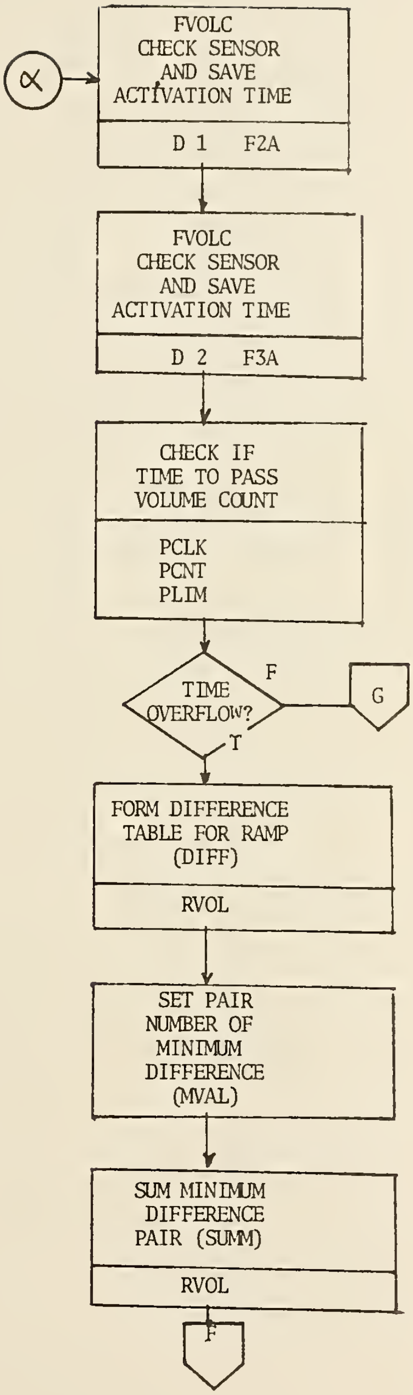

GBVVI
FORMER INITIALIZATION
ADDED INITIALIZATION
EXIT

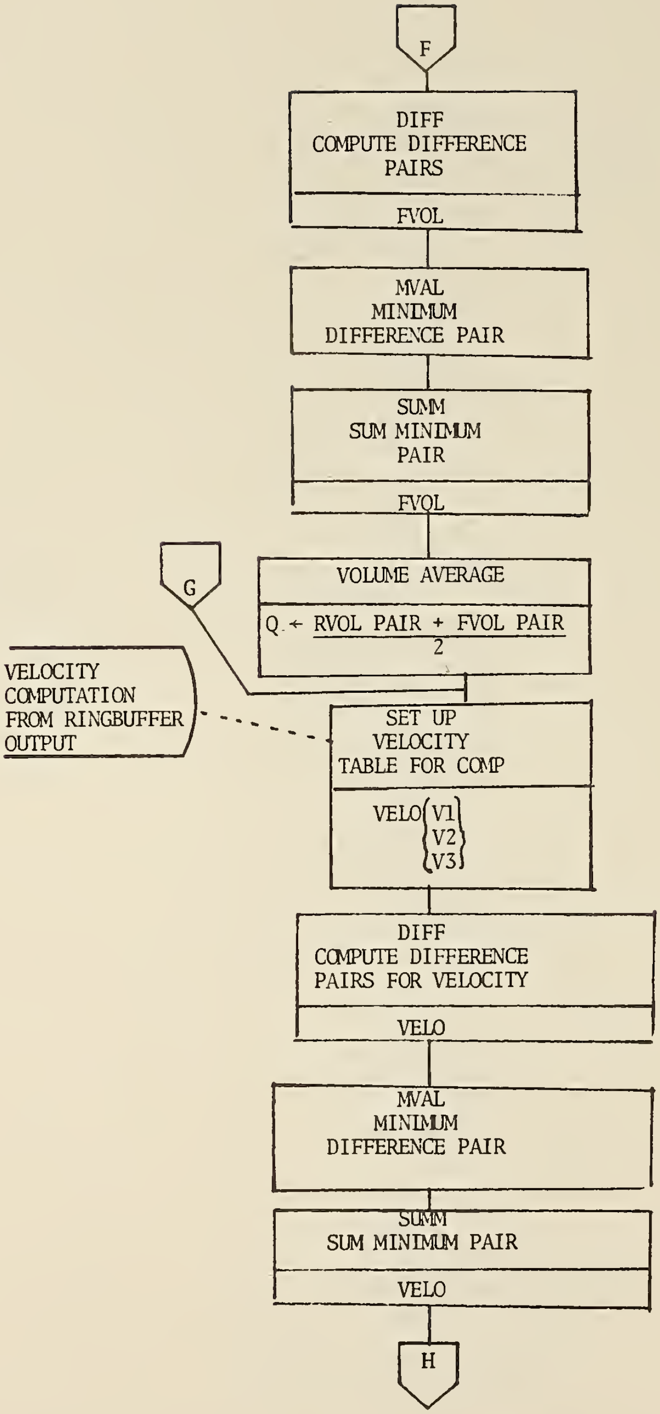

FORMER INITIALIZATION
OLDW
K41W
K42W
K43W
K44W
BEGV
← Ø
EXIT

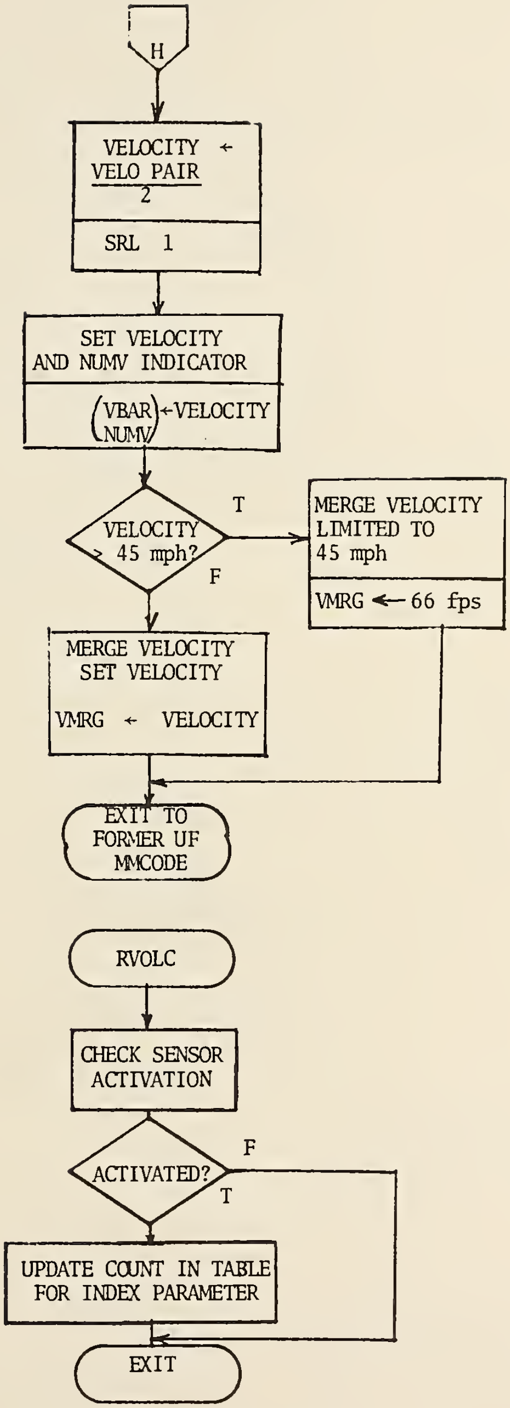

ADDED INITIALIZATION
SENSOR ACTIVATIONS,
COUNTS,
VELOCITIES
← Ø
EXIT

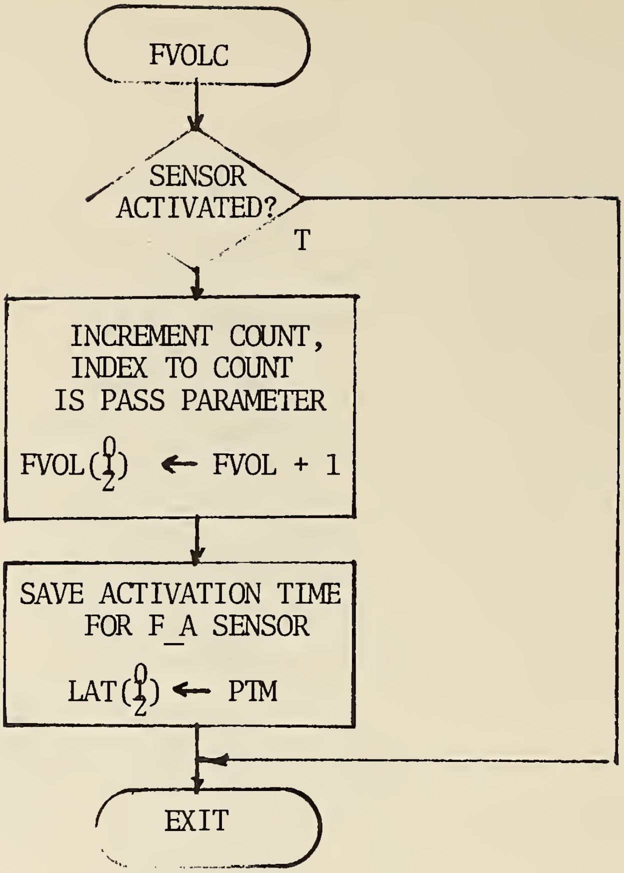

VELVOL
MAIN PROCESSING ENTRY
K30CHK
INPUT SENSOR INFORMATION
SENSOR WORDS 1
2
3
COMPUTE ACTIVATIONS
SWA ← INPUT ∧ HISTORY
SET PRESENT TIME
PTM ← TIME
VELOCITY CHECKS
F_B SENSORS ACTIVATED
CODE REPEATED FOR: F1B
F2B
F3B
(only described once)
E
F_B ACTIVATED ?
F
T
NEXT F_B SENSOR CHECK OR TO COUNT LOGIC (EVP) IF ALL CHECKED
D
COMPUTE TIME ACROSS GAUGE LENGTH OF SENSOR
PTM - F_A ACTIVATION TIME
TIME REASONABLE? (TCHK)
F
T
COMPUTE VELOCITY
GAUGE LENGTH/ TIME
C

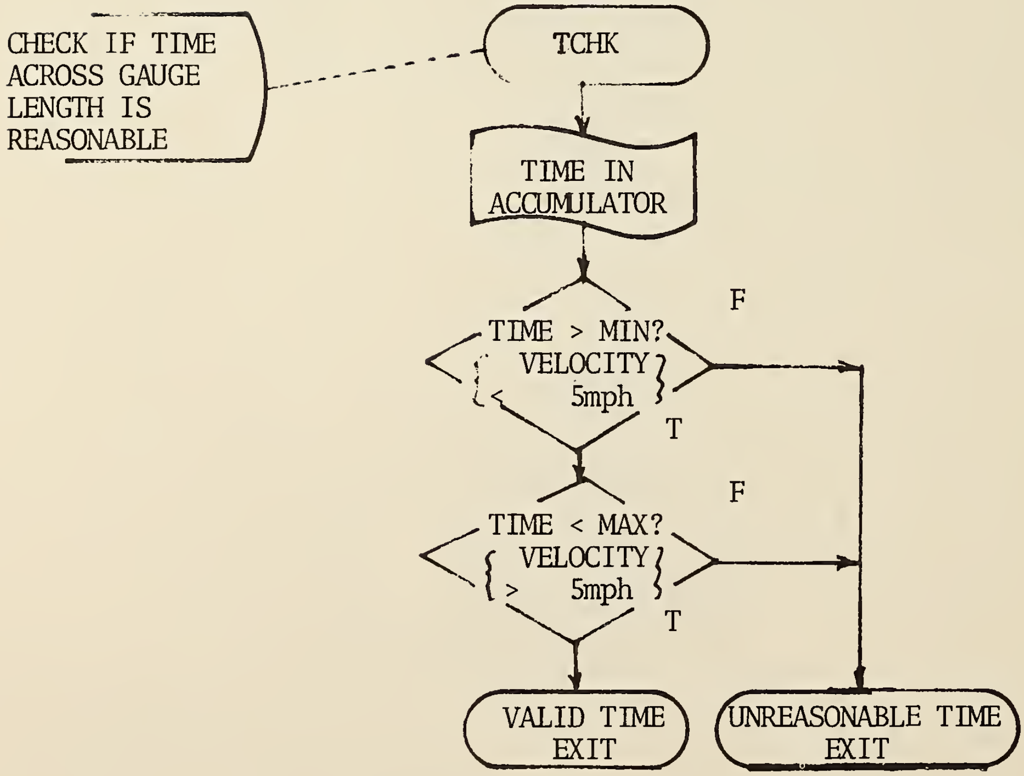

C
RINGBUFFER
VELOCITY TABLE
TOTAL
AVERAGE
E
D
EVP
VOLUME
CHECK
RVOLC
CHECK AND
COUNT SENSOR
D Ø R1
RVOLC
CHECK AND
COUNT SENSOR
D 1 R2
FVOLC
CHECK AND
COUNT SENSOR
D2 R3
FVOLC
CHECK AND
COUNT SENSOR
AND SAVE ACTIVATION
TIME
D Ø F1A
α

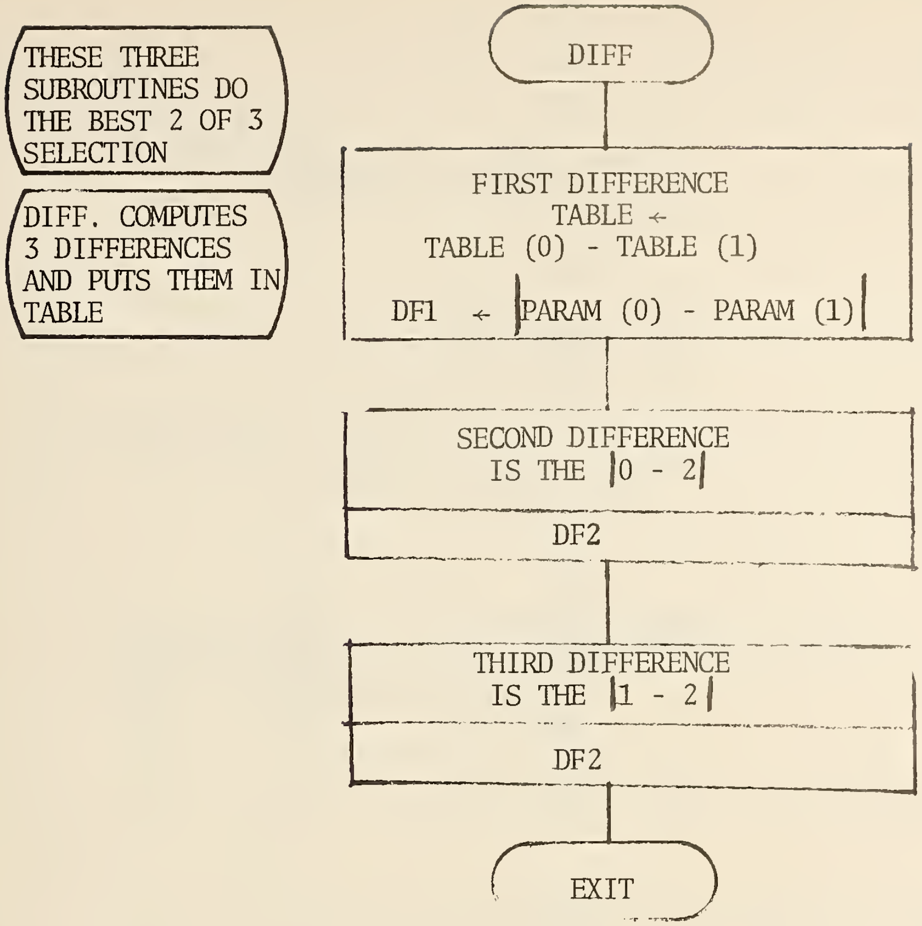

FVOLC
CHECK SENSOR
AND SAVE
ACTIVATION TIME
D 1 F2A
FVOLC
CHECK SENSOR
AND SAVE
ACTIVATION TIME
D 2 F3A
CHECK IF
TIME TO PASS
VOLUME COUNT
PCLK
PCNT
PLIM
TIME
OVERFLOW?
F
G
T
FORM DIFFERENCE
TABLE FOR RAMP
(DIFF)
RVOL
SET PAIR
NUMBER OF
MINIMUM
DIFFERENCE
(MVAL)
SUM MINIMUM
DIFFERENCE
PAIR (SUMM)
RVOL
F

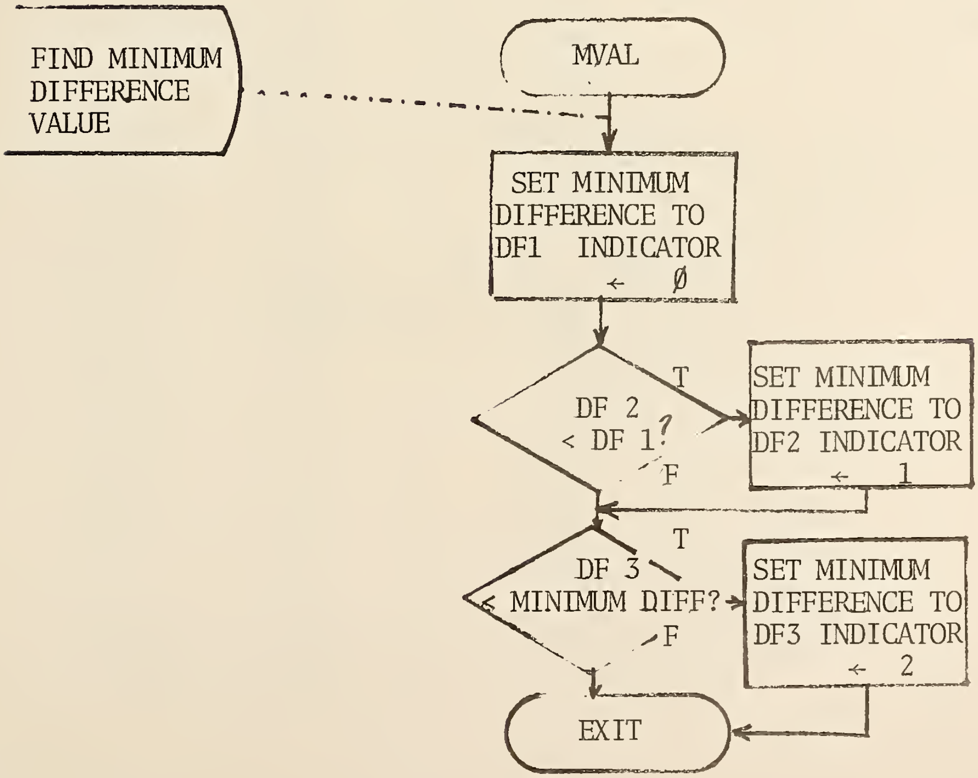

F
DIFF
COMPUTE DIFFERENCE PAIRS
FVOL
MVAL
MINIMUM DIFFERENCE PAIR
SUMM
SUM MINIMUM PAIR
FVOL
VOLUME AVERAGE
Q ← RVOL PAIR + FVOL PAIR / 2
G
VELOCITY COMPUTATION FROM RINGBUFFER OUTPUT
SET UP VELOCITY TABLE FOR COMP
VELO V1 V2 V3
DIFF
COMPUTE DIFFERENCE PAIRS FOR VELOCITY
VELO
MVAL
MINIMUM DIFFERENCE PAIR
SUMM
SUM MINIMUM PAIR
VELO
H

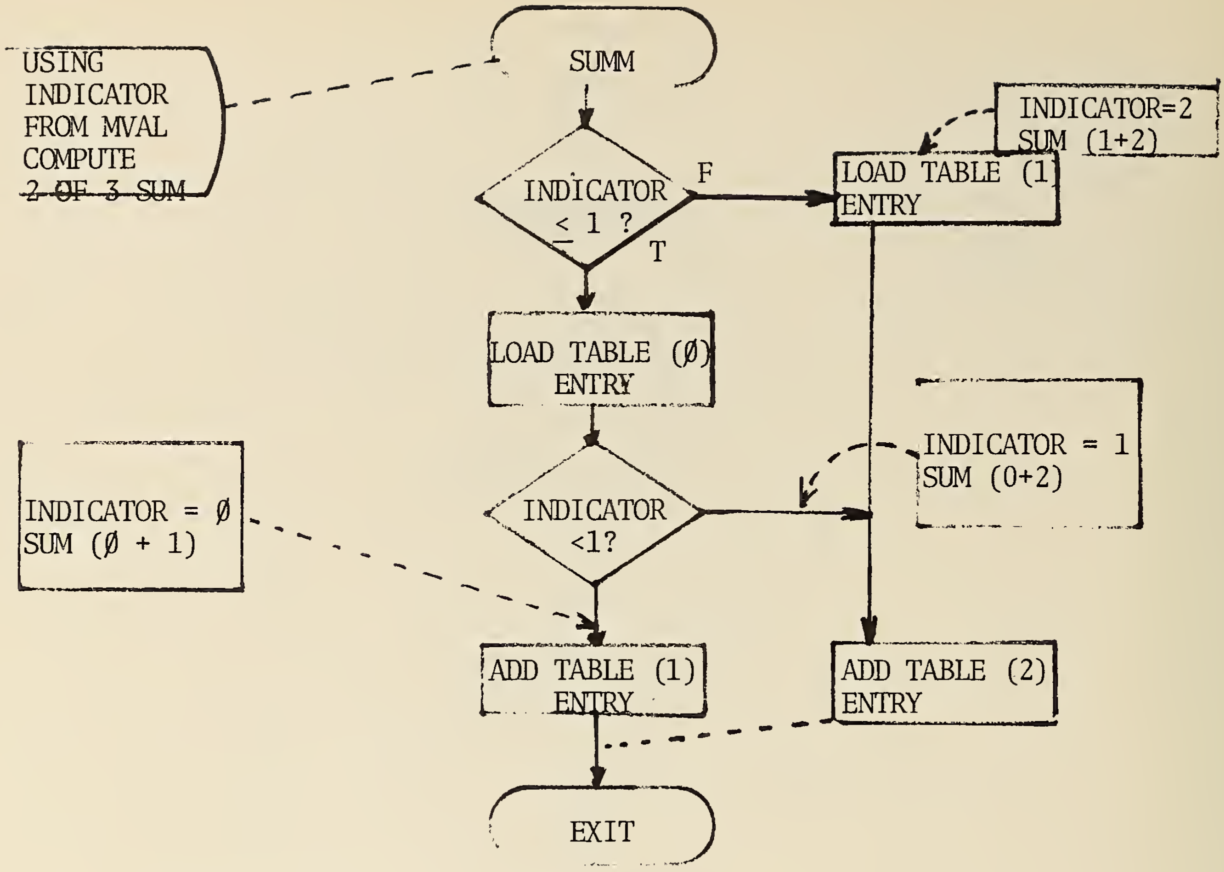

H
VELOCITY ← VELO PAIR / 2
SRL 1
SET VELOCITY AND NUMV INDICATOR
(VBAR NUMV) ← VELOCITY
VELOCITY > 45 mph?
T
F
MERGE VELOCITY LIMITED TO 45 mph
VMRG ← 66 fps
MERGE VELOCITY SET VELOCITY
VMRG ← VELOCITY
EXIT TO FORMER UF MMCODE
RVOLC
CHECK SENSOR ACTIVATION
ACTIVATED?
F
T
UPDATE COUNT IN TABLE FOR INDEX PARAMETER
EXIT

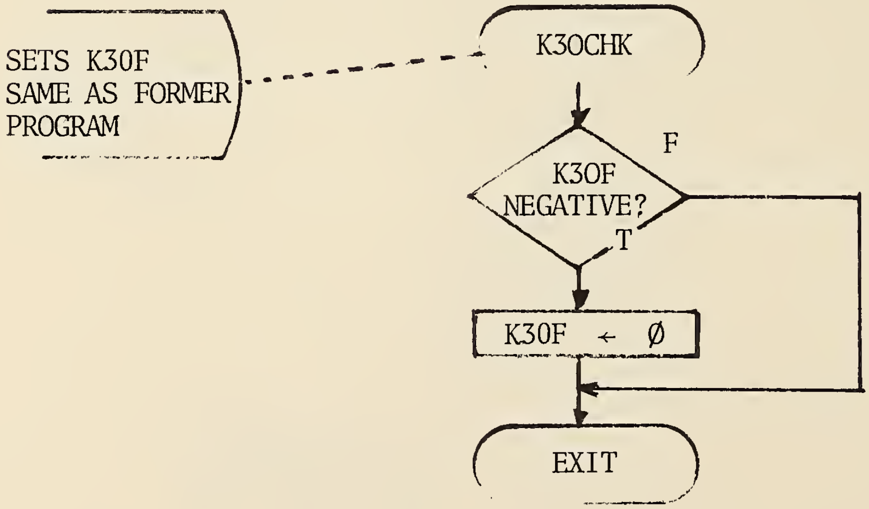

FVOLC
SENSOR ACTIVATED?
T
INCREMENT COUNT, INDEX TO COUNT IS PASS PARAMETER
FVOL(0 1 2) ← FVOL + 1
SAVE ACTIVATION TIME FOR F_A SENSOR
LAT(0 1 2) ← PTM
EXIT


CHECK IF TIME ACROSS GAUGE LENGTH IS REASONABLE
TCHK
TIME IN ACCUMULATOR
TIME > MIN?
{VELOCITY < 5mph}
F
T
TIME < MAX?
{VELOCITY > 5mph}
F
T
VALID TIME EXIT
UNREASONABLE TIME EXIT


THESE THREE SUBROUTINES DO THE BEST 2 OF 3 SELECTION
DIFF. COMPUTES 3 DIFFERENCES AND PUTS THEM IN TABLE
DIFF
FIRST DIFFERENCE
TABLE ←
TABLE (0) - TABLE (1)
DF1 ← |PARAM (0) - PARAM (1)|
SECOND DIFFERENCE
IS THE |0 - 2|
DF2
THIRD DIFFERENCE
IS THE |1 - 2|
DF2
EXIT


FIND MINIMUM DIFFERENCE VALUE
MVAL
SET MINIMUM DIFFERENCE TO DF1 INDICATOR ← Ø
DF 2 < DF 1?
T
F
SET MINIMUM DIFFERENCE TO DF2 INDICATOR ← 1
DF 3 < MINIMUM DIFF?
T
F
SET MINIMUM DIFFERENCE TO DF3 INDICATOR ← 2
EXIT


SUMM
USING
INDICATOR
FROM MVAL
COMPUTE
2 OF 3 SUM
INDICATOR=2
SUM (1+2)
INDICATOR ≤ 1 ?
F
T
LOAD TABLE (1)
ENTRY
LOAD TABLE (Ø)
ENTRY
INDICATOR = 1
SUM (0+2)
INDICATOR = Ø
SUM (Ø + 1)
INDICATOR <1?
ADD TABLE (1)
ENTRY
ADD TABLE (2)
ENTRY
EXIT


K30CHK
SETS K30F
SAME AS FORMER
PROGRAM
K30F NEGATIVE?
F
T
K30F ← Ø
EXIT

```
                 1 ' GB VELOCITY AND VOLUME  #1
                 2 *  VERSION PREPARED 7-21-74
                 3 *  MODIFIED 3-17-77
                 4 *  Q BASED ON TWO OF THREE (F1, F2, F3) PLUS TWO
                 5 *     OF THREE (R1, R2, R3)
                 6 *
                 7 *
                 8 *
                 9            ORIG   X'2000'
                10 *
      180A      11 MINCLK     EQU    X'180A'    ONE MINUTE CLOCK
      180B      12 TNMSCK     EQU    X'180B'    10 MSEC CLOCK
      1803      13 Q          EQU    X'1803'
      0490      14 SPMUL      EQU    X'490'
      04E0      15 SPDIV      EQU    X'4E0'
      1806      16 VBAR       EQU    X'1806'
      1805      17 VMRG       EQU    X'1805'
      1807      18 K30F       EQU    X'1807'    K30 FLAG (F1 IN TAMPA)
      1808      19 K30V       EQU    X'1808'    VEL OF LAST VEH ACROSS F1
                20 *
                21 *
                22 *
      23C9      23 RNGBFR     EQU    X'23C9'     RING BUFFER ROUTINE
      0021      24 TIME       EQU    X'0021'     .002 SEC CLOCK
      0060      25 F1GL       EQU    X'0060'     GAUGE LENGTH
      0070      26 F2GL       EQU    X'0070'
      0080      27 F3GL       EQU    X'0080'
      3EE8      28 CUSC       EQU    X'3EE8'     CHECK SECOND COUNT
                29 *
                30 *
2000 101C       31 JVELVL     JMP    VELVOL   MAIN ENTRY POINT FOR GBVV
2001 0000       32 BIXR       D      0   BEGINNING INDEX TO VELOCITY TABLE
2002 0000       33 CIXR       D      0   CURRENT INDEX TO VELOCITY TABLE
2003 0000       34 TVL        D      0   TOTAL 3 MINUTE VELOCITY
2004 0000       35 NUMV       D      0   NUMBER OF VEHICLES OVER 3 MINUTES
2005 0000       36 OVEL       D      0   OVERTIME VELOCITY SUM
2006 0000       37 MINTIM     D      0   CURRENT MINUTE TIME
2007 0000       38 SECTIM     D      0   CURRENT SECOND TIME
2008 0000       39 OLDW       D      0   OLD STATUS WORD
2009 0000       40 STOT       D      0   VOLUME SUBTOTAL
200A 0000       41 K41W       D      0   CURRENT STATE OF F0-A
200B 0000       42 K42W       D      0   CURRENT STATE OF F0-C
200C 0000       43 K43W       D      0   CURRENT STATE OF F0-E
200D 0000       44 K44W       D      0   CURRENT STATE OF SPARE
200E 0400       45 K41A       D      X'0400'   MASK FOR F0-A
200F 0000       46 K42A       D      X'0000'   MASK FOR F0-C
2010 0000       47 K43A       D      X'0000'   MASK FOR F0-E
2011 0000       48 K44A       D      X'0000'   MASK FOR SPARE
2012            49 VOLT       RES    4   STORAGE FOR VOL TOTAL
2016 0000       50 NEWW       D      0   NEW STATUS WORD
2017 0000       51 BEGV       D      0   BEGIN PTR FOR TABLE
2018 0003       52 ENDV       D      3   END POINTER FOR TABLE
2019 0000       53 TOTE       D      0   CURRENT VOL TOTAL
201A 0000       54 CLOK       D      0   1 MINUTE CLOCK
                55
```

```
                ' GBVELVOL  #2
                *
                * MAIN ROUTINE ENTRY POINT
                *
2018 0000
201C 601B       VELVOL   SUBR
201D 216F                JSX   K30CHK    CHECK FORMER FLAG
                *
                * SENSOR A ACTIVATED--SAVE TIME, COUNT IT
                * SENSOR B ACTIVATED--COMPUTE VELOCITY
                *
                * SENSOR INPUT
                *
201E 02F6                DIN   15,6  SENSOR WORD 1
201F 7188                STW   SWI
2020 02F5                DIN   15,5  WORD 2
2021 7189                STW   SWI+1
2022 02F3                DIN   15,3  WORD 3
2023 718A                STW   SWI+2
                *
                * DETERMINE ACTIVATIONS
                *
2024 9264                LDX   =2
2025 0040                SLM
     2026       DTALP    EQU   $
2026 393B                LDW * SWH   HISTORY, LAST INPUTS
2027 0120                INJ
2028 E938                AND * SWI   WITH PRESENT INPUTS
2029 798E                STW * SWA   GIVES ACTIVATIONS THIS TIME
202A 8938                LDW * SWI   PRESENT INPUT
202B 793B                STW * SWH   BECOMES HISTORY
202C 0501                DXS   1
202D 1026                JMP   DTALP
                *
                * AND PRESENT TIME
                *
202E 0080                SMB   TIME
202F 8021                LDW   TIME
2030 7187                STW   PTM
```

```
                ' GBVELVOL   #3
                *
                * FIRST CHECK SENSORS FOR VELOCITY
                *
  2031 818E                LDW   SWA
  2032 0A12                SLL   2    F1B
  2033 0820                SAM        ACTIVATED?
  2034 104A                JMP   F2M      NO, NEXT SENSOR CHECK
  2035 71A3                STW   SAV   YES,
  2036 8187                LDW   PTM   CALCULATE TIME
  2037 B197                SUB   L1    ACROSS GAUGE LENGTH
  2038 2145                JSX   TCHK  IS IT REASONABLE?
  2039 1049                JMP   F1E   NOPE, IGNORE
  203A 71A2                STW   DELTM  YES, GAUGE LENGTH / DELTA TIME
  203B 0100                CLR
  203C 0081                SPDIV
S 203D 24E0
  203E 0060                D     F1GL    F1 GAUGE
  203F 21A2                D     DELTM
  2040                     RES   1  DIVISION REMAINDER
  2041 0800                SAZ        IF OVERFLOW--INVALID
  2042 1044                JMP   $+2    HAVE SPEED IN FPS
  2043 1049                JMP   F1E
  2044 23C9                JSX   RNGBFR
  2045 21A4                D     VT1
  2046 0000       V1I      D     0     INDEX TO VEL TAB
  2047 0000       V1T      D     0     TOTAL OF 64 VEL'S
  2048 0000       V1A      D     0     AVERAGE VEL
       2049       F1E      EQU   $
  2049 81A3                LDW   SAV
                *
       204A       F2M      EQU   $
  204A 0A12                SLL   2          F2B
  204B 0820                SAM
  204C 1062                JMP   F3M
  204D 71A3                STW   SAV
  204E 8187                LDW   PTM
  204F B198                SUB   L2
  2050 2145                JSX   TCHK
  2051 1061                JMP   F2E
  2052 71A2                STW   DELTM
  2053 0100                CLR
  2054 0081                SPDIV
S 2055 24E0
  2056 0070                D     F2GL
  2057 21A2                D     DELTM
  2058                     RES   1
  2059 0800                SAZ
  205A 105C                JMP   $+2
  205B 1061                JMP   F2E
  205C 23C9                JSX   RNGBFR
  205D 21E4                D     VT2
  205E 0000       V2I      D     0
  205F 0000       V2T      D     0
  2060 0000       V2A      D     0
       2061       F2E      EQU   $
  2061 81A3                LDW   SAV
```

GBVELVOL #3 PAGE 4

GBVELVOL #4 PAGE 5

```
                * GBVELVOL    #4
                *
                * CONTINUE SENSOR CHECK FOR VELOCITY
                *  COMMENTS FOR THIRD SECTION PARALLEL THOSE
                *  FOR THE FIRST SECTION FOR F1B.
                *
       2062     F3M       EQU   $
  2062 0A12               SLL   2            F3B
  2063 0820               SAM
  2064 1078               JMP   EVP
  2065 8187               LDW   PTM
  2066 B199               SUB   L3
  2067 2145               JSX   TCHK
  2068 1078               JMP   EVP
  2069 71A2               STW   DELTM
  206A 0100               CLR
  206B 0081               SPDIV
S 206C 24E0
  206D 0080               D     F3GL
  206E 21A2               D     DELTM
  206F                    RES   1
  2070 0800               SAZ
  2071 1073               JMP   $+2
  2072 1078               JMP   EVP
  2073 23C9               JSX   RNGBFR
  2074 2224               D     VT3
  2075 0000     V3I       D     0
  2076 0000     V3T       D     0
  2077 0000     V3A       D     0
```

```
                ' GBVELVOL   #5
                *
                * SECOND CHECK ACTIVATIONS FOR COUNTS
                *
     2078       EVP        EQU    $
2078 818F                  LDW    SWA+1     RAMP, COUNT ONLY
2079 0A14                  SLL    4
207A 212A                  JSX    RVOLC
207B 0000                  D      0
207C 0A13                  SLL    3
207D 212A                  JSX    RVOLC
207E 0001                  D      1
207F 8190                  LDW    SWA+2
2080 0A43                  SRC    3
2081 212A                  JSX    RVOLC
2082 0002                  D      2
                *
2083 818E                  LDW    SWA     FREEWAY, COUNT AND LAST
2084 0A13                  SLL    3       ACTIVATION TIME
2085 2137                  JSX    FVOLC
2086 0000                  D      0
2087 0A12                  SLL    2
2088 2137                  JSX    FVOLC
2089 0001                  D      1
208A 0A12                  SLL    2
208B 2137                  JSX    FVOLC
208C 0002                  D      2
```

```
                ' GBVELVOL   #6
                *
                * VOLUME COMPUTATION
                *
                *
                *  CHECK IF TIME TO PASS VOLUME
                *
  208D 008F                 CUSC
S 208E 26E8
  208F 0000     PCLK        D      0
  2090 0000     PCNT        D      0
  2091 00B4     PLIM        D      180    3 MINUTES
  2092 1094                 JMP    $+2
  2093 10A9                 JMP    NCP
                *
                * VOLUME AVERAGE
                *
  2094 214D                 JSX    DIFF    VOLUME AVERAGE
  2095 2191                 D      RVOL   RAMP
  2096 2161                 JSX    MVAL
  2097 2178                 JSX    SUMM
  2098 2191                 D      RVOL
  2099 71A3                 STW    SAV
  209A 214D                 JSX    DIFF    VOLUME AVERAGE
  209B 2194                 D      FVOL    FREEWAY
  209C 2161                 JSX    MVAL
  209D 2178                 JSX    SUMM
  209E 2194                 D      FVOL
  209F A1A3                 ADD    SAV    AND TOGETHER THEY GIVE
                *                          YOU LITTLE VOLUMES
  20A0 0A01                 SRL    1
  20A1 0038                 SMB    Q
  20A2 7003                 STW    Q
  20A3 0100                 CLR
  20A4 9265                 LDX    =5
  20A5 0040                 SLM
       20A6     CLVO        EQU    $
  20A6 7991                 STW  * RVOL
  20A7 0501                 DXS    1
  20A8 10A6                 JMP    CLVO
                *
       20A9     NCP         EQU    $
                *
                *VELOCITY AVERAGE
                *
  20A9 8048                 LDW    V1A
  20AA 719A                 STW    V1
  20AB 8050                 LDW    V2A
  20AC 719B                 STW    V2
  20AD 8077                 LDW    V3A
  20AE 719C                 STW    V3
  20AF 214D                 JSX    DIFF
  20B0 219A                 D      VELO
  20B1 2161                 JSX    MVAL
  20B2 2178                 JSX    SUMM
  20B3 219A                 D      VELO
  20B4 0A01                 SRL    1
```

```
20B5 0086           SMB   VBAR   MOVE BACK TO GBS
20B6 7006           STW   VBAR
20B7 7004           STW   NUMV
20B8 F266           CMW   =66   MAX GB SPEED (45 MPH)
20B9 0890           SLE
20BA 8266           LDW   =66
20BB 0086           SMB   VMRG
20BC 7005           STW   VMRG  NO, SET GB VEL AT MERGE PT.
20BD 10D1           JMP   MMCODE   GO DO UF STUFF
              *
              *
              *
20BE                RES   X'20C3'-$

```



```
              ' GBVELVOL   #7
              *
              * INITIALIZATION ROUTINE
              *
20C3 0000
20C4 60C3     GBVVI   SUBR
20C5 0100             CLR
20C6 210F             JSX   OLDINIT  FORMER INITIALIZATION
20C7 2119             JSX   CINIT   ADDED INITIALIZATION
20C8 90C3             EXIT  GBVVI
20C9 2300
              *
              *
              *
20CA                  RES   X'20CC'-$

```

```
                ' GBVELVOL-- UF MM CODE    #8
                *
20CC 0000       UFPAM     D      0
20CD 0000       UFPLA     D      0
20CE 0BB8       UFPBL     D      3000    264FT.*500/44FT PER SEC.
20CF 0000       UFVMRG    D      0
20D0 002C       UF44      D      44
                *
20D1 0056       MMCODE    SMB    VMRG
20D2 8005                 LDW    VMRG
20D3 F0CF                 CMW    UFVMRG
20D4 0870                 SNE
20D5 10DE                 JMP    OUTCODE
20D6 70CF                 STW    UFVMRG
20D7 E0D0                 SUB    UF44
20D8 0130                 CAX
20D9 0040                 SLM
20DA 88E0                 LDW *  PAMTBL
20DB 70CC                 STW    UFPAM
20DC 88F7                 LDW *  PLATBL
20DD 70CD                 STW    UFPLA
20DE 901B       OUTCODE   LDX    VELVOL-1
20DF 2800                 JSX *  0
20E0 19BF       PAMTBL    DATA   6591
20E1 1888                 DATA   6280
20E2 175A                 DATA   5978
20E3 1636                 DATA   5686
20E4 151B                 DATA   5403
20E5 1408                 DATA   5128
20E6 12FC                 DATA   4860
20E7 11F8                 DATA   4600
20E8 10FA                 DATA   4346
20E9 1003                 DATA   4099
20EA 0F12                 DATA   3858
20EB 0E27                 DATA   3623
20EC 0D41                 DATA   3393
20ED 0C60                 DATA   3168
20EE 0B84                 DATA   2948
20EF 0AAD                 DATA   2733
20F0 09DA                 DATA   2522
20F1 090C                 DATA   2316
20F2 0841                 DATA   2113
20F3 077A                 DATA   1914
20F4 06B7                 DATA   1719
20F5 05F8                 DATA   1528
20F6 053A                 DATA   1338
20F7 0000       PLATBL    DATA   0
20F8 00A7                 DATA   167
20F9 014D                 DATA   333
20FA 01F4                 DATA   500
20FB 029B                 DATA   667
20FC 0341                 DATA   833
20FD 03E8                 DATA   1000
20FE 048F                 DATA   1167
20FF 0535                 DATA   1333
2100 05DC                 DATA   1500
2101 0683                 DATA   1667
```

GBVELVOL-- UF MM CODE #8 PAGE 11

```
2102 0729            DATA   1833
2103 07D0            DATA   2000
2104 0877            DATA   2167
2105 091D            DATA   2333
2106 09C4            DATA   2500
2107 0A6B            DATA   2667
2108 0B11            DATA   2833
2109 0BB8            DATA   3000
210A 0C5F            DATA   3167
210B 0D05            DATA   3333
210C 0DAC            DATA   3500
210D 0E53            DATA   3667
```

GBVELVOL COLLECTED SUBROUTINES #9 PAGE 12

```
           ' GBVELVOL COLLECTED SUBROUTINES  #9
           *
           * OLD INITIALIZATION ROUTINE
           *
210E 0000
210F 610E    OLDINIT  SUBR
2110 7008             STW   OLDW
2111 700A             STW   K41W
2112 700B             STW   K42W
2113 700C             STW   K43W
2114 700D             STW   K44W
2115 7017             STW   BEGV
2116 910E             EXIT  OLDINIT
2117 2800
           *
           * ADDED INITIALIZATION
           *
2118 0000
2119 6118    CINIT    SUBR
211A 9267             LDX   =21
211B 0040             SLM
     211C    CINLP1   EQU   $
211C 7988             STW * SWI
211D 0501             DXS   1
211E 111C             JMP   CINLP1
211F 708F             STW   PCLK
2120 7090             STW   PCNT
2121 9046             LDX   V1I
2122 79A4             STW * VT1
2123 905E             LDX   V2I
2124 79E4             STW * VT2
2125 9075             LDX   V3I
2126 7A24             STW * VT3
2127 9118             EXIT  CINIT
2128 2800
```

```
                ' GBVELVOL SUBROUTINES CONT'D   #10
                *
                * RAMP VOLUME COUNT
                *
2129 0000
212A 6129       RVOLC     SUBR
212B 0820                 SAM
212C 2801                 JSX * 1
212D 71A3                 STW   SAV
212E 9800                 LDX * 0
212F 0040                 SLM
2130 8991                 LDW * RVOL
2131 A268                 ADD   =1
2132 7991                 STW * RVOL
2133 81A3                 LDW   SAV
2134 9129                 EXIT  RVOLC,1
2135 2801
                *
                * FREEWAY VOLUME COUNT AND LAST ACTIVATION TIME
                *
2136 0000
2137 6136       FVOLC     SUBR
2138 0820                 SAM
2139 2801                 JSX * 1
213A 71A3                 STW   SAV
213B 9800                 LDX * 0
213C 0040                 SLM
213D 8994                 LDW * FVOL
213E A268                 ADD   =1
213F 7994                 STW * FVOL
2140 8187                 LDW   PTM
2141 7997                 STW * LAT
2142 81A3                 LDW   SAV
2143 9136                 EXIT  FVOLC,1
2144 2801
                *
                * TIME CHECK FOR VELOCITY RANGE
                *
     2145       TCHK      EQU   $
2145 F1A1                 CMW   MIN
2146 0880                 SGR
2147 2800                 JSX * 0   BAD
2148 F1A0                 CMW   MAX
2149 0840                 SLS
214A 2800                 JSX * 0   BAD
214B 2801                 JSX * 1   OK
```

```
                ' GBVELVOL SUBROUTINES CONT'D  #11
                *
                * DIFFERENCE TABLE
                *
2140 0000
214D 614C       DIFF      SUBR
214E 9800                 LDX * 0
214F 8800                 LDW * 0
2150 B801                 SUB * 1
2151 0810                 SAP
2152 0110                 CMP
2153 719D                 STW   DF1     0 - 1
2154 8800                 LDW * 0
2155 B802                 SUB * 2
2156 0810                 SAP
2157 0110                 CMP
2158 719E                 STW   DF2    0 - 2
2159 8801                 LDW * 1
215A B802                 SUB * 2
215B 0810                 SAP
215C 0110                 CMP
215D 719F                 STW   DF3    1 - 2
215E 914C                 EXIT  DIFF,1
215F 2801
                *
                * MINIMUM VALUE IN TABLE
                *
2160 0000
2161 6160       MVAL      SUBR
2162 9269                 LDX   =0
2163 819D                 LDW   DF1    0 - DF1
2164 F19E                 CMW   DF2    1 - DF2
2165 0880                 SGR
2166 1169                 JMP   MVL10 2 - DF3
2167 819E                 LDW   DF2
2168 9268                 LDX   =1
     2169       MVL10     EQU   $
2169 F19F                 CMW   DF3
216A 0840                 SLS
216B 9264                 LDX   =2
216C 0140                 CXA
216D 9160                 EXIT  MVAL
216E 2800
```

```
GBVELVOL SUBROUTINES CONT'D  #12                         PAGE  15

                'GBVELVOL SUBROUTINES CONT'D  #12
                *
                * REPLACES K30(F1) HANDLING FROM FORMER VELVOL
                *
     216F       K30CHK    EQU   $
216F 0086                 SMB   K30F
2170 8007                 LDW   K30F    IF FLAG -VE
2171 0820                 SAM
2172 2800                 JSX * 0
2173 0100                 CLR        CLEAR IT
2174 0086                 SMB   K30F
2175 7007                 STW   K30F
2176 2800                 JSX * 0


GBVELVOL SUBROUTINES CONT'D  #13                         PAGE  16

                ' GBVELVOL SUBROUTINES CONT'D  #13
                *
                * SUM TWO OF THREE
                *
2177 0000
2178 6177       SUMM      SUBR
2179 9800                 LDX * 0
217A F268                 CMW   =1
217B 0880                 SGR
217C 117F                 JMP   SMLE
217D 8801                 LDW * 1
217E 1184                 JMP   SMEQ
     217F       SMLE      EQU   $
217F 8800                 LDW * 0
2180 0840                 SLS
2181 1184                 JMP   SMEQ
2182 A801                 ADD * 1
2183 1185                 JMP   SMEXT
     2184       SMEQ      EQU   $
2184 A802                 ADD * 2
     2185       SMEXT     EQU   $
2185 9177                 EXIT  SUMM,1
2186 2801
```

```
                ' GBVELVOL DATA STORAGE  #14
                *
                *
                *
2187            PTM      RES    1      TIME, .002 SEC
                *
2188            SWI      RES    3      SENSOR STORAGE
218B            SWH      RES    3
218E            SWA      RES    3
                *
                *
2191            RVOL     RES    0      VOLUME COUNT RAMP
2191            R01      RES    1       R1
2192            R02      RES    1       R2
2193            R03      RES    1       R3
                *
2194            FVOL     RES    0      VOLUME COUNT FREEWAY
2194            F01      RES    1       F1A
2195            F02      RES    1       F2A
2196            F03      RES    1       F3A
                *
                *
2197            LAT      RES    0      LAST ACTIVATION TIME
2197            L1       RES    1       F1A
2198            L2       RES    1       F2A
2199            L3       RES    1       F3A
                *
219A            VELO     RES    0      VELOCITY
219A            V1       RES    1       F1
219B            V2       RES    1       F2
219C            V3       RES    1       F3
                *
                *
219D            DIFFR    RES    0      DIFFERENCE TABLE
219D            DF1      RES    1
219E            DF2      RES    1
219F            DF3      RES    1
                *
                *
21A0 0553       MAX      D      1363   5 MPH, .002 SEC COUNTS
21A1 0035       MIN      D      53     128 MPH, 20 FT. GAUGE
21A2            DELTM    RES    1
21A3            SAV      RES    1
                *
21A4            VT1      RES    64
21E4            VT2      RES    64
2224            VT3      RES    64
                *
                *
                *
                         END

2264 0002
2265 0005
2266 0042
2267 0015
2268 0001
```

2269 0000

NO ERRORS

BEGV	2017	BIXR	2001	CINIT	2119	CINLP1	211C
CIXR	2002	CLOK	201A	CLVO	20A6	CUSC	3EE8
DELTM	21A2	DF1	219D	DF2	219E	DF3	219F
DIFF	214D	DIFFR	219D	DTALP	2026	ENDV	2018
EVP	2078	F1E	2049	F1GL	0060	F2E	2061
F2GL	0070	F2M	204A	F3GL	0080	F3M	2062
F01	2194	F02	2195	F03	2196	FVOL	2194
FVOLC	2137	GBVVI	20C4	0VELVL	2000	K30CHK	216F
K30F	1807	K30V	1808	K41A	200E	K41W	200A
K42A	200F	K42W	200B	K43A	2010	K43W	200C
K44A	2011	K44W	200D	L1	2197	L2	2198
L3	2199	LAT	2197	MAX	21A0	MIN	21A1
MINCLK	180A	MINTIM	2006	MMCODE	20D1	MVAL	2161
MVLI0	21690	NCP	20A9	NEWW	2016	NUMV	2004
OLDINIT	210F	OLDW	2008	OUTCODE	20DE	OVEL	2005
PAMTBL	20E0	PCLK	208F	PCNT	2090	PLATBL	20F7
PLIM	2091	PTM	2187	Q	1803	RNGBFR	23C9
R01	2191	R02	2192	R03	2193	RVOL	2191
RVOLC	212A	SAV	21A3	SECTIM	2007	SMEQ	2184
SMEXT	2185	SMLE	217F	SPDIV	04E0	SPMUL	0490
STOT	2009	SUMM	2178	SWA	218E	SWH	218B
SWI	2188	TCHK	2145	TIME	0021	TNMSCK	180B
TOTE	2019	TVL	2003	UF44	20D0	UFPAM	20CC
UFPBL	20CE	UFPLA	20CD	UFVMRG	20CF	V1	219A
V1A	2048	V1I	2046	V1T	2047	V2	219B
V2A	2060	V2I	205E	V2T	205F	V3	219C
V3A	2077	V3I	2075	V3T	2076	VBAR	1806
VELO	219A	VELVOL	201C	VMRG	1805	VOLT	2012
VT1	21A4	VT2	21E4	VT3	2224		

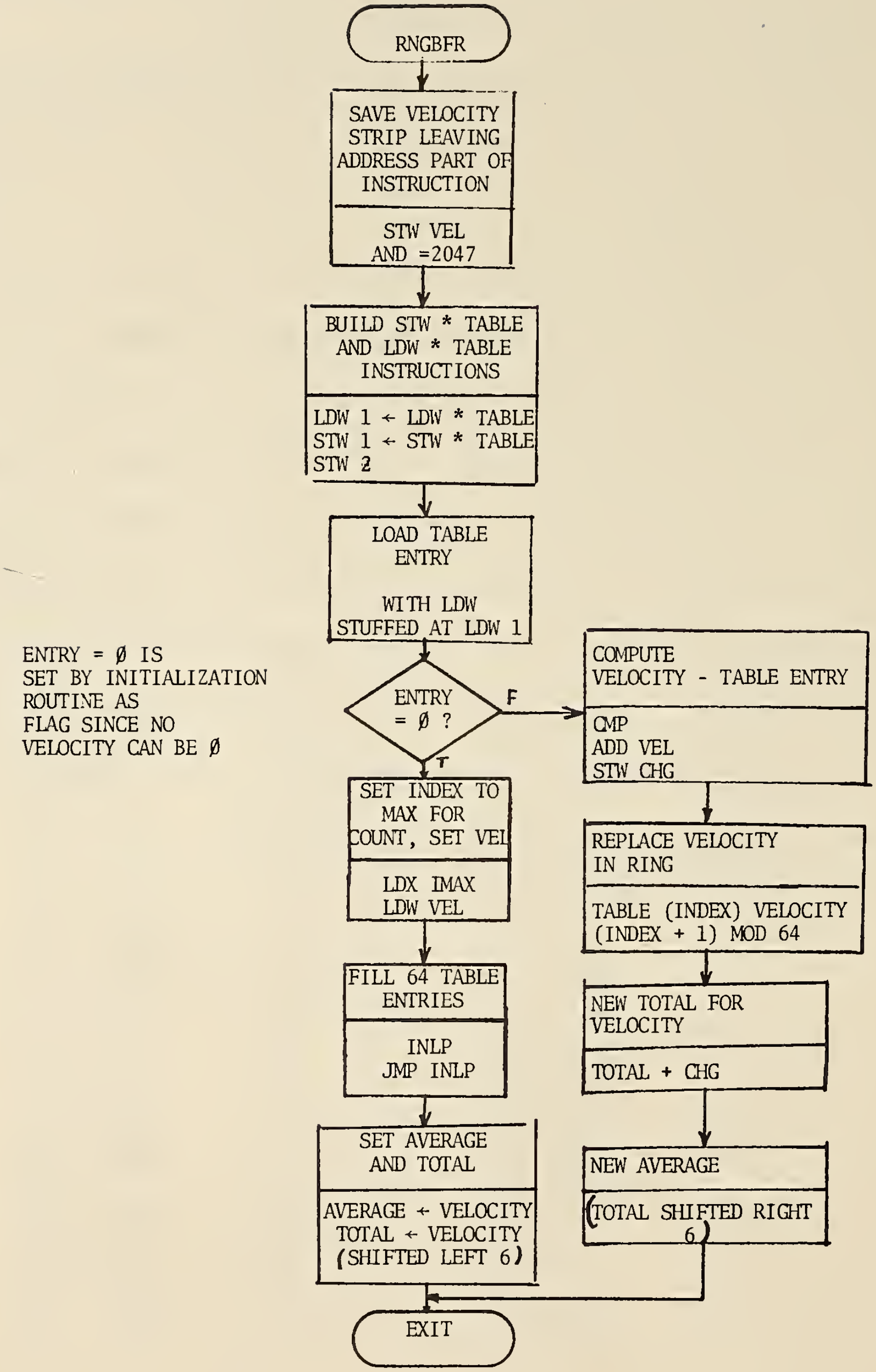
PAS?

RINGBUFFER SUBROUTINE

The subroutine is used by Velocity and Volume Subprogram to do the velocity averages at the highway sensors.

In the call are the address of the 64-word buffer, the total maintained for the buffer, the average and the last index value used to access table.

The entry point is RNGBFR. First, store and load instructions are built to access the table in an indexed-local mode. Then the table entry is retrieved and checked for zero, indicating initialization. For buffer update BUPDT, the difference between the value passed in the accumulator is computed and saved. Then the entry in the ring is replaced with the passed value, the total adjusted by the difference, CHG, and the total shifted right 6 (divide by 64) to give the average. For initialization, the ring is filled with the passed value, the average set to the value, and the total is set to the value shifted left 6 (multiplied by 64).


RNGBFR
SAVE VELOCITY STRIP LEAVING ADDRESS PART OF INSTRUCTION
STW VEL
AND =2047
BUILD STW * TABLE AND LDW * TABLE INSTRUCTIONS
LDW 1 ← LDW * TABLE
STW 1 ← STW * TABLE
STW 2
LOAD TABLE ENTRY
WITH LDW STUFFED AT LDW 1
ENTRY = Ø IS SET BY INITIALIZATION ROUTINE AS FLAG SINCE NO VELOCITY CAN BE Ø
ENTRY = Ø ?
F
T
COMPUTE VELOCITY - TABLE ENTRY
CMP
ADD VEL
STW CHG
SET INDEX TO MAX FOR COUNT, SET VEL
LDX IMAX
LDW VEL
REPLACE VELOCITY IN RING
TABLE (INDEX) VELOCITY
(INDEX + 1) MOD 64
FILL 64 TABLE ENTRIES
INLP
JMP INLP
NEW TOTAL FOR VELOCITY
TOTAL + CHG
SET AVERAGE AND TOTAL
AVERAGE ← VELOCITY
TOTAL ← VELOCITY
(SHIFTED LEFT 6)
NEW AVERAGE
(TOTAL SHIFTED RIGHT 6)
EXIT

```
            ' RING BUFFER  #1
            *
            *       CALL:
            *        RNGBFR
            *(0)     D      ADDR 64-WORD BUFFER, SAME WORD PAGE
            *                 AS RNGBFR.
            *(1)     D      0-RELATIVE INDEX TO BUFFER
            *(2)     RES    1   TOTAL, 64 WORDS
            *(3)     RES    1   AVERAGE
            *(4)     RETURN
            *
            *  ACCUMULATOR--REPLACEMENT VALUE
            *
            * TO INITIALIZE--SET BUFFER ENTRY = 0
            *      BUFFER FILLED WITH REPLACEMENT VALUE
            *
            * IN SAME WORD PAGE AS VELVOL DUE
            * TO REQUIRED TABLE LOCALITY.
            *
                     ORIG  X'23C8'  ANCHOR POINT
            *
            *
            *
23C8 0000
23C9 63C8   RNGBFR   SUBR
23CA 73FA            STW   VEL
23CB 8800            LDW * 0    BUILD LDW * AND STW *
23CC E3F5            AND   A2047    INSTRUCTION FOR LOCAL TABLE
23CD C3F8            ORI   STWI   ADDRESSING
23CE 73D9            STW   STW1
23CF 73E6            STW   STW2
23D0 A3F6            ADD   A4096
23D1 73D4            STW   LDW1
23D2 9801            LDX * 1   FIRST, SET INDEX SO WE CAN
23D3 0040            SLM
23D4 0A10   LDW1     NOP      PICK UP TABLE ENTRY
23D5 0800            SAZ     SKIP IF INITIALIZATION
23D6 13E2            JMP   BUPDT   OTHERWISE DO UPDATE
```

```
      RING BUFFER  #2                                         PAGE   2

                 ' RING BUFFER  #2
                 *
                 * INITIALIZE RING BUFFER
                 *
23D7 93F9                LDX   IMAX   INITIALIZE WORKS
23D8 83FA                LDW   VEL    TO REPLACEMENT VALUE
     23D9         INLP   EQU   $
23D9 0A10         STW1   NOP          FILLING RING
23DA 0501                DXS   1
23DB 13D9                JMP   INLP
                 *
23DC 93C8                LDX   RNGBFR-1    THEN THE AVERAGE
23DD 0050                SGM
23DE 7803                STW * 3
23DF 0A16                SLL   6   TOTAL = 64 * REPLACEMENT
23E0 7802                STW * 2
23E1 2804                JSX * 4   RETURN -- DONE

      RING BUFFER   #3                                        PAGE   3

                 ' RING BUFFER   #3
                 *
                 * UPDATE AVERAGE AND BUFFER
                 *
     23E2         BUPDT    EQU   $
23E2 0110                  CMP
23E3 A3FA                  ADD   VEL
23E4 73E8                  STW   CHG
23E5 83FA                  LDW   VEL   PUT REPLACEMENT VALUE IN RING
23E6 0A10         STW2     NOP
23E7 0401                  IXS   1   INCREMENT RING INDEX
23E8              CHG      RES   1
23E9 0140                  CXA
23EA E3F9                  AND   IMAX  KEEP MODULO 64
23EB 0050                  SGM
23EC 93C8                  LDX   RNGBFR-1
23ED 7801                  STW * 1
23EE 8802                  LDW * 2
23EF A3E8                  ADD   CHG
23F0 7802                  STW * 2
23F1 A3F7                  ADD   A32  TO BE ROUNDED AFTER SHIFT
23F2 0A06                  SRL   6
23F3 7803                  STW * 3
23F4 2804                  JSX * 4
```

```
                ' RING BUFFER    #4
                *
                * DATA AREA
                *
23F5 07FF       A2047      D       2047
23F6 1000       A4096      D       4096
23F7 0020       A32        D       32
23F8 7800       STWI       STW *   0
23F9 003F       IMAX       D       63
23FA            VEL        RES     1

```



```
                ' RING BUFFER    #5
                *
                *
                *
                           END
```

NO ERRORS


```
A2047     23F5  A32       23F7  A4096     23F6  BUPDT     23E2
CHG       23E8  IMAX      23F9  INLP      23D9  LDWI      23D4
RNGBFR    23C9  STWI      23D9  STW2      23E6  STWI      23F8
VEL       23FA
PAS?
```

BAND TRIM SUBROUTINE

Band Trim is called by GBUD to smooth the display before letting it go to the field. The program is modeled after a FORTRAN simulation of August 15, 1976, which was a smoothing algorithm with no band back-up.

Entry is at ROUTINE which some mode checks are done. No smoothing is done unless:

1. Mode is moving merge (MODE = 3)
2. The time has elapsed since last smoothing to permit another (ELT in units of .002 sec.),
3. And the system is unmasked (K51F system mask = -1 and PSF power save mask = -1).

For the system masked, a clear display is sent and for any mode other than moving merge, nothing is done to display.

The trim of bands starts at TRIMR, with a comparison of the present requested output and what was last output to field. If they are the same no more is done and the display is left to go to field. An "exclusive or" function detects changes and saves it in a work buffer, WB.

When band changes are detected control passes to BNDCHGS. A scan is made across the buffer looking for changes to process (WB contains changes.) The scan is entered whenever a change has been completely processed and the end of the display has not been found. At end of the display, it goes to COPY which moves the bands to the field buffer (DBUF).

To determine the trimming necessary on a band end, information about the change is necessary:

1. The first light (FCHG) of the change in the new band (DBUF),
2. The light immediately preceding change (BFR) in the history or old band (OB), and
3. The light immediately following change (AFT) in the old band (OB).

The light positions in the band are set then the status of each light is obtained:

FCLT, first change light,

BFLT, before light, and

AFLT, after light.

If a band has the end of the display as a terminuns then BFR or AFT is set to -1 to indicate it, and LCHG is the length of the change.

Then the change is processed on a case - by - case basis, with Ø for light off and 1 for light on:

	BFLT	AFLT	FCLT	BAND ACTION INDICATED
Non end cases	0	0	0	Turn band off, permit
	0	0	1	Trying to create a band, not allowed
	1	1	1	Band filling in a gap, allow one light advance of trailing gap end
	1	1	0	A cut-in has taken place, gap creation allowed
	0	1	1	Trailing band end back-up, not allowed
	0	1	0	Trailing band end advance, if it is greater than 14 treat as cut-in otherwise allow single light advance

	BFLT	AFLT	FCLT	BAND ACTION INDICATED
Non end cases (cont.)	1	0	1	Heading band end advance, allow single light advance.
	1	0	0	Heading band end back-up, allow only for greater than 14 which is a cut-in.
End cases	BEGIN	END	1	Turn whole band on, allow single light advance
	BEGIN	1	1	Trailing band end back-up, not allowed
	BEGIN	1	0	Trailing band end advance, if greater than 2 treat as cut-in otherwise allow single light advance
	BEGIN	0	1	Heading band end back-up, only allow if greater than 2 cut-in
	0	END	1	Trailing band end back-up, not allowed
	0	END	0	Trailing band end advance, if greater than 2 assume cut-in otherwise single light advance
	1	END	1	Heading band end advance, allow single light advance
	1	END	0	Heading band end back-up, if greater than 2 assume cut-in otherwise not allowed.

The band change has 2 actions:

1. Single light change of state and
2. Multiple light change of state.

Both changes set a range for the change loop (CHNGRLP) to do then branch to it.

For either no band change or change completed the routine checks to see if a band piece exists in the rest of the word the band change may have overlapped into, then returns to search for more band changes or goes to COPY to put new band out. (NZWD or ZWD).

See the report text for a detailed flow chart of this algorithm.

CHANGES TO BAND TRIMMER SOURCE CODE
NOT APPEARING ON ASSEMBLY LISTING

The changes indicated below are incorporated into the final versions of the source and binary paper tapes for the BANDTRIM programs as of April 22, 1977.

1.) Replace source code lines 49 and 50 with 5 new lines:

```
        SNE
        JMP    MMOOD
        JSX    CLARET   NOT RIGHT, CLEAR HISTORY
        JMP    REXIT    AND DEPART
MMOOD   EQU    $
```

2.) Insert the following statement after line 65 of the current code:

```
REXIT   EQU    $
```

3.) Replace line 80 of the current code with:

```
        JSX    CLARET   CLEAR OLD HISTORY
        JMP    COPY     RUN IT OUT TO FIELD
CLARET  SUBR
```

4.) Replace line 87 of the current code with:

```
        EXIT   CLARET   HISTORY CLEAR FINISHED
```

```
              'BAND TRIMMER  #1
              *
              *  CALL:
              *        BNDTRM
              *
              *  PROCESSES THE BANDS TO BE DISPLAYED TO
              * THE FIELD ACCORDING TO FORTRAN SIMULATION
              * OUTPUT FOR AUG 25, 1976.
              *
              *
              *  EXTERNAL REFERENCES
              *
     1802     MODE      EQU   X'1802'   SYSTEM MODE
     0001     SM        EQU   1
     0002     SG        EQU   2
     0003     MM        EQU   3
     0829     DBUF      EQU   X'829'   GREEN BAND OUTPUT BUFFER
     0021     TIME      EQU   X'21'    SYSTEM TIME WORD, .002 SEC
     15E3     LITR      EQU   X'15E3'
     08B1     RM        EQU   X'08B1'  BORROW MASKS FROM GBUD
     180C     K51F      EQU   X'180C'  MASK FLAG
     180D     PSF       EQU   X'180D'  POWER SAVE MASK
              *
                        ORIG  X'24B9'
     24B9     BNDTRM    EQU   $
24B9 14D9               JMP   ROUTINE
              *
              * DATA USED LOCALLY TO PROGRAM
              *
24BA          LST       RES   1  SAVED TIME
24BB 0019     ELT       D     25 = .05 SEC/.002 SEC, INTERVAL
24BC 0002     BCUTIN    D     2  TUBES OFF = CUT-IN AT BEG
24BD 000F     ICUTIN    D     15 TUBES OFF = CUT-IN WITHIN
24BE 0002     ECUTIN    D     2  TUBES OFF = CUT-IN AT END
24BF          OB        RES   10  OLD, HISTORY GREEN BAND
24C9          WB        RES   10  BAND CHANGES WORK AREA
24D3          SC        RES   1   SCRATCH
24D4          BFLT      RES   1   LIGHT BEFORE CHANGE
24D5          AFLT      RES   1   LIGHT AFTER CHANGE
24D6          FCLT      RES   1   FIRST LIGHT IN CHANGE
24D7          LCHG      RES   1   LENGTH OF BAND CHANGE
```

```
               'BANDTRIM MAIN PROCESSING  #2
               *
24D8 0000
24D9 64D8      ROUTINE   SUBR
24DA 0086                SMB   MODE  NO DIDDLING UNLESS
24DB 8002                LDW   MODE IN THE RIGHT MOOD
24DC F5E2                CMW   =MM
24DD 0860                SEQ
24DE 14F7                JMP   CLARET NOT RIGHT, DON'T MESS WITH IT
24DF 0080                SMB   TIME   SET SYSTEM TIME
24E0 8021                LDW   TIME
24E1 B4BA                SUB   LST  -SAVE TIME = ELAPSED TIME
24E2 F4BB                CMW   ELT   SKIP IF ELAPSED INTERVAL
24E3 0840                SLS
24E4 14EE                JMP   TMOUT
     24E5      COPY      EQU   $
24E5 95E3                LDX   =9  MOVE OLD TO DBUF
24E6 0040                SLM
     24E7      NCHG      EQU   $
24E7 8CBF                LDW * OB
24E8 0082                SMB   DBUF
24E9 7829                STW * DBUF
24EA 0501                DXS   1
24EB 14E7                JMP   NCHG
24EC 94D8                EXIT  ROUTINE
24ED 2800
               *
               *  UPDATE TIME ELAPSED, MODE MUST BE MM
               *
     24EE      TMOUT     EQU   $
24EE 0080                SMB   TIME
24EF 8021                LDW   TIME  SYSTEM TIME
24F0 74BA                STW   LST  TO SAVED TIME
24F1 0086                SMB   K51F  BANDS MASKED?
24F2 800C                LDW   K51F   LET'S FIND OUT
24F3 0086                SMB   PSF  INCLUDE POWER SAVE MASK
24F4 E00D                AND   PSF
24F5 0800                SAZ         THEY ARE, CLEAN OUT OLD
24F6 14FE                JMP   TRIMR  NO MASK, TRIM AWAY
     24F7      CLARET    EQU   $
24F7 95E3                LDX   =9
24F8 0040                SLM
24F9 0100                CLR
24FA 7CBF                STW * OB
24FB 0501                DXS   1
24FC 14FA                JMP   $-2
24FD 14E5                JMP   COPY     CLEAN DISPLAY TO FIELD
               *
               *  BAND TRIM
               *
     24FE      TRIMR     EQU   $
24FE 0100                CLR
24FF 74D3                STW   SC  INITIALIZE CHECK
               *
               *  FIRST XOR DBUF BAND AND OLD BAND
               *
2500 95E3                LDX   =9
```

```
2501 0040          SLM
     2502    XORLP EQU  $
2502 0082          SME  DBUF
2503 3E29          LDW * DBUF  GET GREEN BAND TO GO
2504 DCBF          ORE * OB  EXCLUSIVE OR WITH OLD
2505 7CC9          STW * WB  SAVE IT IN WORK
2506 C4D3          ORI  SC  ACCUMULATE OVERALL CHANGE
2507 74D3          STW  SC
2508 0501          DXS  1
2509 1502          JMP  XORLP
250A 84D3          LDW  SC  ANY CHANGES?
250B 0800          SAZ       NOPE, DON'T WASTE TIME
250C 150F          JMP  BNDCHGS  GO DO THEM
250D 94D8          EXIT ROUTINE
250E 2800
```

```
              113 'BANDTRIM, CHANGES TO PROCESS  #3
              114 *
     250F     115 BNDCHGS   EQU   $
250F 0100     116           CLR    CLEAR LIGHT NUMBER
2510 7522     117           STW   CLNO
2511 0130     118           CAX       AND SET CHANGE INDEX
              119 *
              120 * LOCATE A NONZERO CHANGE WORD
              121 *
2512 0040     122           SLM
     2513     123 ZWD       EQU   $
2513 3CC9     124           LDW * WB     CHANGES IN WORK BUFFER
2514 0300     125           SAZ       NO CHANGE
2515 151F     126           JMP   NZWD  FOUND SOME
2516 3522     127           LDW   CLNO  KICK COUNT UP TO NEXT WORD
2517 A5E4     128           ADD   =16
2518 F5E5     129           CMW   =144
2519 0890     130           SLE
251A 14E5     131           JMP   COPY   OLD BAND TO DBUF
251B 7522     132           STW   CLNO
     251C     133 INCW      EQU   $
251C 0401     134           IXS   1
251D          135 AD        RES   1   LAST LIGHT TO CHANGE
251E 1513     136           JMP   ZWD
              137 *
              138 * FIND LIGHT THAT CHANGED
              139 *
     251F     140 NZWD      EQU   $
     251F     141 STLZ      EQU   $
251F 0085     142           LITR
S 2520 25E3
2521 24C9     143           D     WB
2522          144 CLNO      RES   1
2523 0000     145           D     0
2524 0810     146           SAP      SKIP IF NO CHANGE
2525 152A     147           JMP   CHGD
2526 3522     148           LDW   CLNO  GO TO NEXT CHANGE TEST
2527 A5E6     149           ADD   =1
2528 7522     150           STW   CLNO
2529 151F     151           JMP   STLZ
              152 *
              153 * FOUND FIRST CHANGE BIT
              154 *
     252A     155 CHGD      EQU   $
252A 3522     156           LDW   CLNO   FOR DECISION
252B 7559     157           STW   FCHG   NEED FIRST CHANGED
252C A5E6     158           ADD   =1
252D 7535     159           STW   CON
252E 7553     160           STW   AFT
252F 7522     161           STW   CLNO
2530 B5E7     162           SUB   =2  AND BEFORE AND AFTER
2531 754A     163           STW   BFR   LIGHTS IN OLD BUFFER
              164 *
              165 * GET END OF CHANGE
              166 *
     2532     167 LECHG     EQU   $
2532 0085     168           LITR
```

```
S 2533 25E3
  2534 2409   169         D     #B
  2535        170 CON     RES   1
  2536 0000   171         D     0
  2537 0ED0   172         SAM     STILL IN CHANGE
  2538 1544   173         JMP   ECHG   FOUND END
  2539 2535   174         LDA   CON
  253A A5E6   175         ADD   =1
  253B F5EC   176         CMW   =160
  253C 0E40   177         SLS
  253D 1542   178         JMP   EDNL
  253E 7522   179         STW   CLNO
  253F 7535   180         STW   CON
  2540 7553   181         STW   AFT
  2541 1532   182         JMP   LDCNG
              183
```

```
                'BANDTRIM, END OF CHANGE  #4
                *
                * END OF BAND FOUND
                *
       2542     EBND      EQU    $
  2542 85E9               LDW    =-1     MARK IT
  2543 7553               STW    AFT
                *
                * END OF CHANGE, DETERMINE LIGHT STATES
                *
       2544     ECHG      EQU    $
  2544 854A               LDW    BFR
  2545 0810               SAP
  2546 154D               JMP    BGBND
  2547 0085               LITR
S 2548 25E3
  2549 24BF               D      OB
  254A          BFR       RES    1
  254B 0000               D      0
  254C 74D4               STW    BFLT
       254D     BGBND     EQU    $
  254D 8553               LDW    AFT
  254E 0810               SAP
  254F 1556               JMP    AFTBND
  2550 0085               LITR
S 2551 25E3
  2552 24BF               D      OB
  2553          AFT       RES    1
  2554 0000               D      0
  2555 74D5               STW    AFLT
       2556     AFTBND    EQU    $
  2556 0085               LITR
S 2557 25E3
  2558 0829               D      DBUF
  2559          FCHG      RES    1
  255A 0000               D      0
  255B 74D6               STW    FCLT
                *
                * NOW DECIDE WHAT TO DO, DETERMINE CHANGE LENGTH
                *
  255C 8553               LDW    AFT     AFT = -1 WHEN END OF DISPLAY
  255D 0810               SAP
  255E 85E8               LDW    =160     SET ACCORDINGLY
  255F B54A               SUB    BFR    COMPUTE CHANGE LENGTH
  2560 B5E6               SUB    =1
  2561 74D7               STW    LCHG
```

```
               'BANDTRIM, DETERMINE THE TRIM  #5
               *
               * HAVE THE NECESSARY INFO TO CHECK TRIM
               *   LIGHTS FRAMING CHANGE(BFLT, AFLT), OB
               *   FIRST LIGHT OF CHANGE(FCLT), DBUF
               *   LENGTH OF CHANGE(LCHG)
               *
               * CHECK IF BEGIN OR END INVOLVED
               *
2562 854A              LDW   BFR
2563 C553              ORI   AFT
2564 0810              SAP    SKIP IF NEITHER INCLUDED
2565 1589              JMP   ENDSIN
               *
               * PROCESS FOR NO ENDS INCLUDED
               *
2566 84D4              LDW   BFLT
2567 C4D5              ORI   AFLT
2568 0810              SAP
2569 156E              JMP   OM  (1,1) (0,1) (1,0) CASES
256A 84D6              LDW   FCLT  (0,0) CASE
256B 0820              SAM
256C 15BA              JMP   MLTCHG  (0,0,0) KILL BAND
256D 15CB              JMP   NOBCHG  (0,0,1) NO BAND CREATION
     256E      OM      EQU   $
256E E4D4              AND   BFLT
256F E4D5              AND   AFLT
2570 0820              SAM       SKIP IF (1,1) CHANGE
2571 1576              JMP   OZM
2572 84D6              LDW   FCLT  PROCESS(1,1) CHANGE
2573 0810              SAP         SKIP IF 0
2574 1536              JMP   STCHG  GO PROCESS(1,1,1)
2575 15BA              JMP   MLTCHG  HONOR CUT-IN (1,1,0)
```

```
BANDTRIM, CONTINUE CASES  #6                                   PAGE  8

                'BANDTRIM, CONTINUE CASES  #6
                *
                * LEADING AND TRAILING EDGES
                * (1,0) AND (0,1) CASES
                *
     2576       0ZM       EQU   $
2576 84D5                 LDW   AFLT
2577 0820                 SAM
2578 1581                 JMP   0Z  GO PROCESS (1,0)
2579 84D6                 LDW   FCLT  PROCESS (0,1)
257A 0810                 SAP
257B 15CB                 JMP   NOBCHG  NO BACK-UP (0,1,1)
257C 84D7                 LDW   LCHG  (0,1,0)
257D F4BD                 CMW   ICUTIN    >= 15  EXTINGUISH
257E 0840                 SLS
257F 15BA                 JMP   MLTCHG  ALLOW CUT-IN
2580 15B6                 JMP   STCHG   ADVANCE TRAILING EDGE
     2581       0Z        EQU   $
2581 84D6                 LDW   FCLT    PROCESS (1,0)
2582 0810                 SAP
2583 15B6                 JMP   STCHG  LEADING EDGE ADVANCE (1,0,1)
2584 84D7                 LDW   LCHG    (1,0,0)
2585 F4BD                 CMW   ICUTIN     >= 15 EXTINGUISH
2586 0840                 SLS
2587 15BA                 JMP   MLTCHG   ALLOW CUT-IN
2588 15CB                 JMP   NOBCHG   NO BACK-UP

BANDTRIM, END CASES  #7                                        PAGE  9

                'BANDTRIM, END CASES  #7
                *
                * BEGIN FIRST
                *
     2589       ENDSIN    EQU   $
2589 854A                 LDW   BFR   CHECK WHOLE BAND
258A E553                 AND   AFT
258B 0210                 SAP
258C 15B6                 JMP   STCHG  ALL TURN ON?--ALLOW 1
258D 854A                 LDW   BFR
258E 0820                 SAN
258F 15A3                 JMP   ENDPROC   GO TO END PROCESSING
2590 84D5                 LDW   AFLT
2591 0820                 SAM
2592 159B                 JMP   BZM   GO TO (BEGIN,0) PROCESSING
2593 84D6                 LDW   FCLT  PROCESS (BEGIN,1)
2594 0810                 SAP
2595 15CB                 JMP   NOBCHG  NO BACK-UP (BEGIN,1,1)
2596 84D7                 LDW   LCHG  (BEGIN,1,0)
2597 F4BC                 CMW   BCUTIN   >2 EXTINGUISH
2598 0830                 SGR
2599 15B6                 JMP   STCHG   TRAILING EDGE ADVANCE
259A 15BA                 JMP   MLTCHG   HONOR CUT-IN
     259B       BZM       EQU   $
259B 84D6                 LDW   FCLT   PROCESS (BEGIN,0)
259C 0810                 SAP
259D 15B6                 JMP   STCHG   SPOOL IN
                *                           ADVANCE (BEGIN,0,1)
259E 84D7                 LDW   LCHG   (BEGIN,0,0)
259F F4BC                 CMW   BCUTIN     >2 EXTINGUISH
25A0 0830                 SGR
25A1 15CB                 JMP   NOBCHG  NO BACK-UP
25A2 15BA                 JMP   MLTCHG   HONOR CUT-IN
```

```
          'BANDTRIM, END CASES  #8
          *
          * END CASE
          *
     25A3 ENDPROC  EQU   $
25A3 84D4          LDW   BFLT
25A4 0810          SAP
25A5 15AE          JMP   OEY  GO PROCESS (1,END)
25A6 84D6          LDW   FOLT  PROCESS (0,END)
25A7 0810          SAP
25A8 15CB          JMP   NOBCHG  NO BACK-UP (0,END,1)
25A9 84D7          LDW   LCHG  (0,END,0)
25AA F4DE          CMW   EQUTIN  >2 EXTINGUISH
25AB 08F0          SGR
25AC 15B6          JMP   STCHG  ADVANCE TRAILING EDGE
25AD 15BA          JMP   MLTCHG  HONOR CUT-IN
     25AE OEY      EQU   $
25AE 84D6          LDW   FOLT  PROCESS (1,END)
25AF 0810          SAP
25B0 15B6          JMP   STCHG  LEADING EDGE ADVANCE (1,END,1)
25B1 84D7          LDW   LCHG  (1,END,0)
25B2 F4DE          CMW   EQUTIN  > 2 EXTINGUISH
25B3 08F0          SGR
25B4 15CB          JMP   NOBCHG  NO BAND BACK-UP
25B5 15BA          JMP   MLTCHG  HONOR CUT-IN
```

```
                 'BANDTRIM, CHANGE HANDLING ROUTINE  #9
                 *
                 * TWO CASES OF CHANGES:
                 *   SWITCH STATE OF FIRST TUBE IN CHANGE
                 *   SWITCH STATE OF ALL TUBES IN CHANGE
                 *
     25B6        STCHG     EQU   $
25B6 8559                  LDW   FCHG  SINGLE LIGHT CHANGE
25B7 75C2                  STW   BC    RANGE ONE LIGHT
25B8 751D                  STW   AD
25B9 15BF                  JMP   CHNGR
                 *
     25BA        MLTCHG    EQU   $
25BA 8559                  LDW   FCHG  MULTIPLE LIGHT CHANGE
25BB 75C2                  STW   BC    START WITH FIRST IN CHANGE
25BC A4D7                  ADD   LCHG   AND GO TO LAST
25BD B5E6                  SUB   =1
25BE 751D                  STW   AD
                 *
     25BF        CHNGR     EQU   $
     25BF        CHNGRLP   EQU   $
25BF 0085                  LITR
S 25C0 25E3
                 *
25C1 24BF                  D     0L
25C2             BC        RES   1
25C3 FFFF                  D     -1
25C4 85C2                  LDW   BC
25C5 F51D                  CMW   AD
25C6 0840                  SLS
25C7 15CB                  JMP   CHNGOVER
25C8 A5E6                  ADD   =1
25C9 75C2                  STW   BC
25CA 15BF                  JMP   CHNGRLP
                 *
     25CB        CHNGOVER  EQU   $
     25CB        NOBCHG    EQU   $
```

```
                386 'BANDTRIM, CHECK FOR END OF CHANGES  #10
                387 *
                388 * CHANGE PROCESSED
                389 *
25CB 0100       390            CLR
25CC 0130       391            CAX
25CD 8553       392            LDW    AFT   CHECK FOR CHANGE AT END
25CE 0810       393            SAP
25CF 14E5       394            JMP    COPY
25D0 0A7C       395            SLC D  12
25D1 0A54       396            SLC    4
25D2 74D3       397            STW    SC
25D3 0040       398            SLM
25D4 8CC9       399            LDW *  WB
25D5 94D3       400            LDX    SC
25D6 0082       401            SMB    RM    BORROWING FROM GBUD MASKS,HINT
25D7 E8B1       402            AND *  RM
25D8 0800       403            SAZ
25D9 151F       404            JMP    NZWD
25DA 8553       405            LDW    AFT
25DB F5E5       406            CMW    =144
25DC 0840       407            SLS
25DD 14E5       408            JMP    COPY
25DE 0A04       409            SRL    4
25DF A5E6       410            ADD    =1
25E0 0130       411            CAX
25E1 1513       412            JMP    ZWD
                413.           END

25E2 0003
25E3 0009
25E4 0010
25E5 0090
25E6 0001
25E7 0002
25E8 00A0
25E9 FFFF


NO ERRORS
```

```
AD        251D  AFLT      24D5  AFT       2553  AFTBND    2556
BC        25C2  BCUTIN    24BC  BFLT      24D4  BFR       254A
BGBND     254D  BNDCHGS   250F  BNDTRM    24E9  BZM       259E
CHGD      252A  CHNGOVER  25CB  CHNGR     25BF  CHNGRLP   253F
CLARET    24F7  CLNO      2522  CON       2535  COPY      24E5
DBUF      0829  EBND      2542  ECHG      2544  ECUTIN    24BE
ELT       24BB  ENDPROC   25A3  ENDSIN    2589  FCHG      2559
FCLT      24D6  ICUTIN    24BD  INCW      251C  K51F      180C
LCHG      24D7  LECHG     2552  LITR      15E3  LST       24BA
MLTCHG    25BA  MM        0003  MODE      1802  NCHG      24E7
NOBCHG    25CB  NZWD      251F  OB        24BF  OEM       25AE
OM        256E  OZ        2581  OZM       2576  PSF       180D
RM        08B1  ROUTINE   24D9  SC        24D3  SG        0002
SM        0001  STCHG     25B6  STLZ      251F  TIME      0021
TMOUT     24EE  TRIMR     24F2  WB        24C9  XORLP     2502
ZWD       2513
PAS?
```

FAULT MONITOR

The changes to the Fault Monitor were:

1. Changing the error type-outs on sensors to error checking for critical combinations, taking the system down if they occur and typing information messages every 15 minutes for noncritical combinations,
2. Option to suppress type-outs through OCIS set flag, and
3. Counts on sensors for an hour and since last re-initialization to be printed on request through OCIS or on the hour if type-outs are not suppressed then clear hour counts.

In initialization, the clock for each of the timed checks and type-outs is reset, fault tables cleared and counts zeroed. All the message type-outs which were of fixed length and stored in descending order are now stored as string with markers ':' to indicate a break point (all messages are printed in terms of how many break points to scan over, without printing break characters; '/' indicates carriage return, line feed when imbedded in a type-out). This facilitates any addition of messages or modifications of present ones. The byte address of the message is passed to the print routine which has the former PRRG entry for a time-stamped type-out and a new entry point, TYPO, for a line without a time-stamp.

At PRRG, the message address is saved and the former time conversion routine used. Calls are issued to TYPO to output the data; it uses the same code that PRRG formerly used to output a single character of data with the added checks for ':' and '/'. Whenever a break, ':', is encountered (PNC, pick up next character), BCNT is decremented and when it goes negative,

the print is terminated. Whenever a '/' is encountered, the end of line sequence is output, as are 2 carriage returns and a line feed. OUTCHAR sets a flag and issues the output data command.

Clear display, CDISP, was changed to put out the data to the SCU with successive instructions instead of loop which built the DOT instruction each time.

At SMØZ, the former control program which checked the sensors based on information provided by Highway Vehicle Processor was removed and replaced by calls to:

1. Processor sensor information into tables (SNIP),
2. Check for critical sensor faults (CFAULT), and
3. Check for noncritical sensor faults (NCFAULT).

Then the sensor count flag (CTFG) is checked to determine if the operator requested a type-out. If he did, the flag is cleared and the counts printed (PRCNT).

The inhibit flag (ITYP) is checked before entering the information type-out section at UFØZ and if it is nonzero, no type-outs are issued.

The message section was rewritten to allow the new PRRG/TYPO routines to use them and the headings for counts and fault information added. The sensor tables and count data areas were added for use in the sensor checks.

All the subroutines to do the new functions were added at the former program end. They will be described.

Print Sensor Table (PST) prints 4 words of sensor information, a character for each sensor indicating whether sensor is all right ('Ø') or faulty ('F'). First the loop is set up to get four words and a LDW* instruction is built to access the sensor information in local mode.

PST1Ø is the loop point for getting the new word of sensor data and PSTZØ is the inner loop to print the contents of the sensor word. When 16 characters have been printed, the inner loop passes control back to the outer loop which calls TYPO to put out the carriage return, line feed, then checks for more sensor words.

The Print Sensor Count (PRCNT) prints the count tables. First, a row of headings is printed; second, the counts for current hour printed; and third the counts since initialization printed. The freeway, ramp, and merge areas are printed. PCNTR is called to print the numerical data.

Convert counts to decimal and print them is done by PCNTR. It builds a LDW* instruction to pick values from the table of counts and uses the former conversion routines to produce a decimal value of the count. The number of counts to convert and address of the first count are passed to the routine.

Critical sensors are checked in CFAULT. The sensor inputs (SWI) are "anded" with a stuck on list, inverted and "anded" with a struck off list. If CTIM seconds have elapsed then the tables are checked for the following combinations of sensors:

1.	R3 and R4	yield sign
2.	F4A-F1B	freeway
3.	R6B or R6A	stop light
4.	M1 - M5	merge area
5.	R11	power save mask
6.	R1, R2, R3 (2 of 3)	count

If any of these combinations have been detected, the system resets the field and halts printing out the faulty sensor list. If all of a group

like the freeway, ramp or merge area indicate bad, it is assumed due to inactivity and ignored. The lists of sensor data are reinitialized and routine exited.

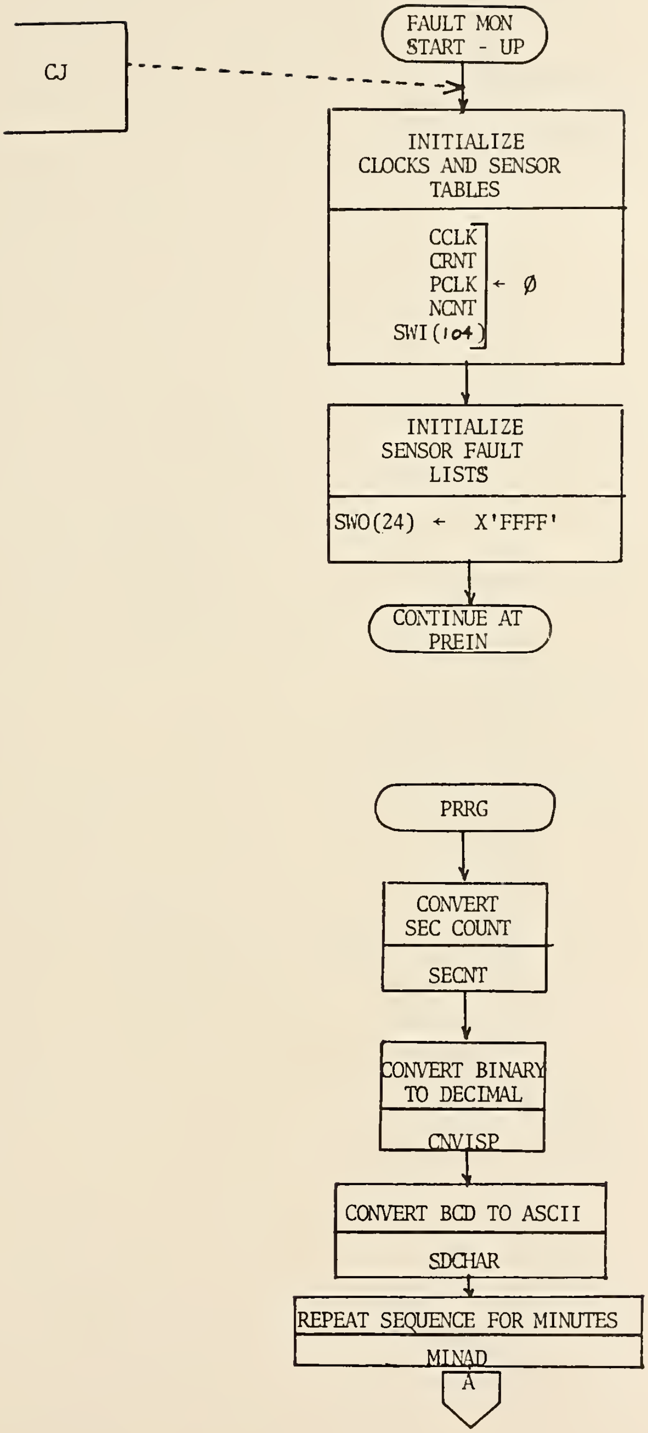
The noncritical sensors are checked in NCFAULT. These will not take the system down and are typed every NTIM seconds as information (if ITYP flag is not set to inhibit type-outs). The sensor lists are printed using PST and reinitialized.

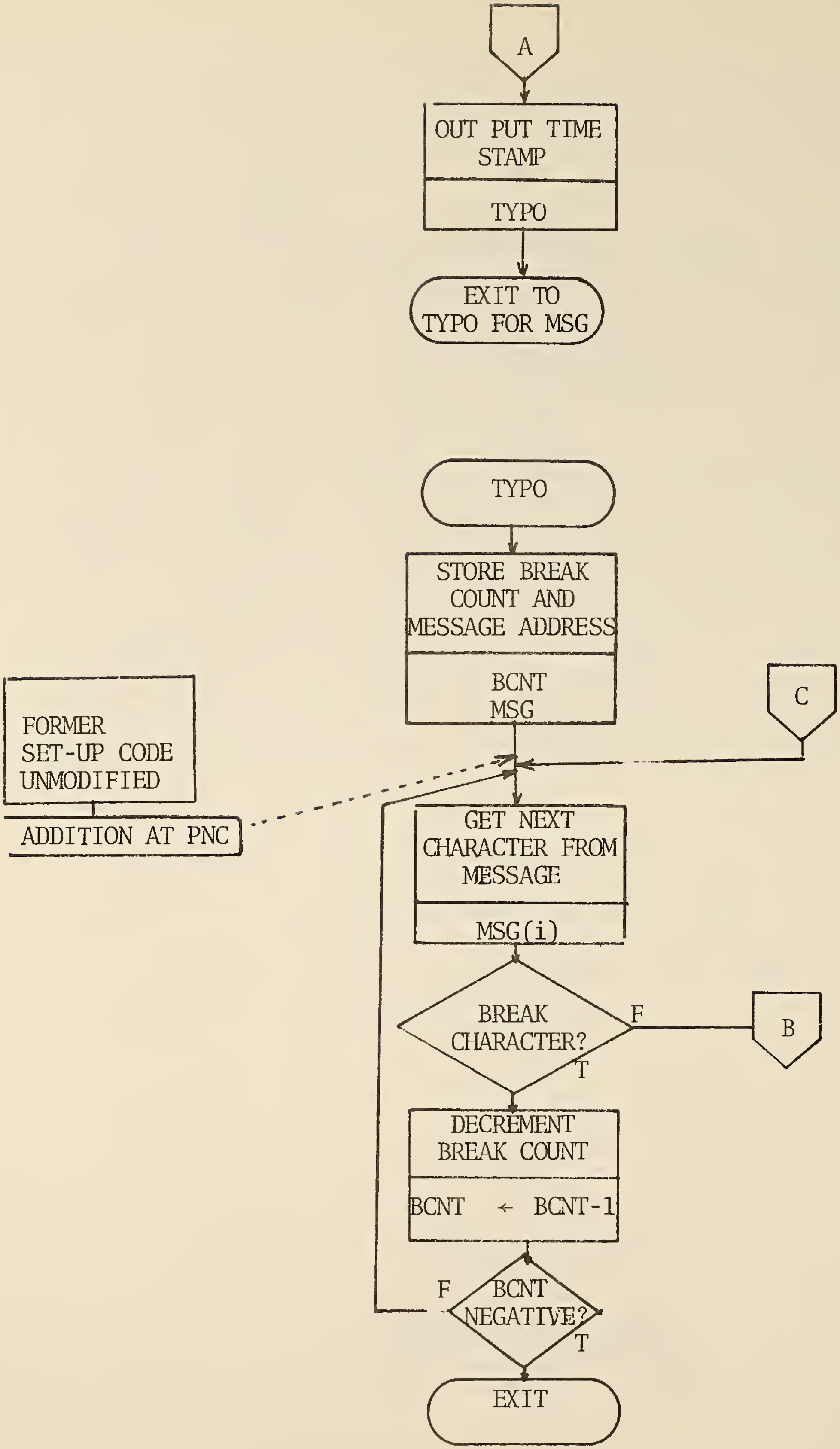
SINP does the sensor input processing for the counts and faults. STP calls for each sensor word input update, the stuck-on, stuck-off, activation and update history lists. Then, if ATIM seconds have elapsed, the counts are typed and zeroed (if allowed by ITYP flag). Then the activation lists are scanned and counts updated using FIT to update freeway counts, RIT to update ramp counts and MIT to update merge area counts.

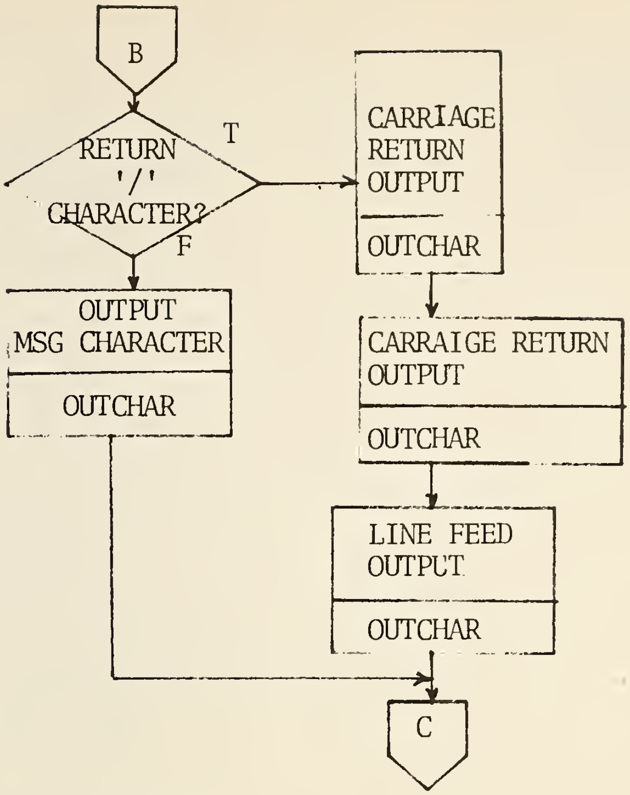
FIT checks if sensor is activated this round and updates count specified in call parameters (index to freeway count table) if it was.

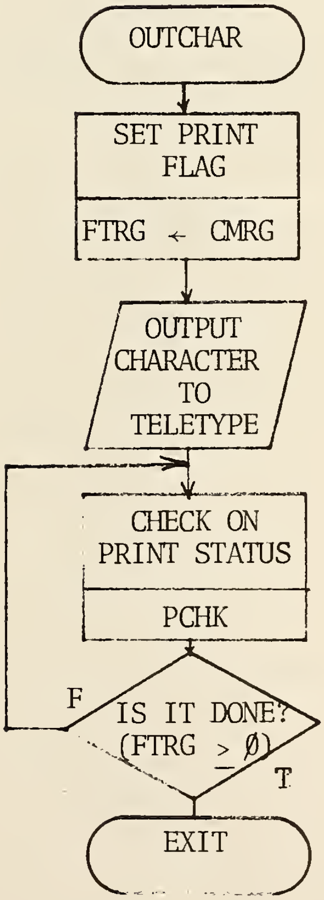
RIT does the similar function to ramp count table.

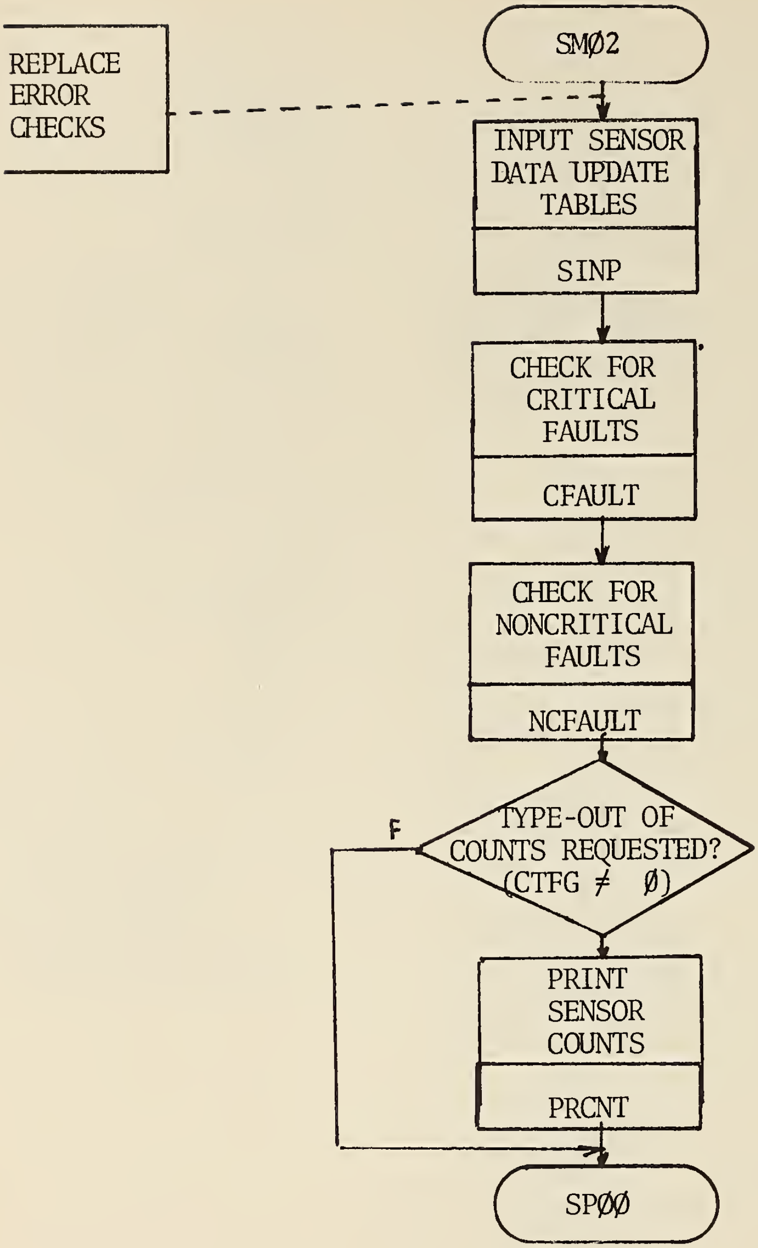
MIT does it for the merge area.

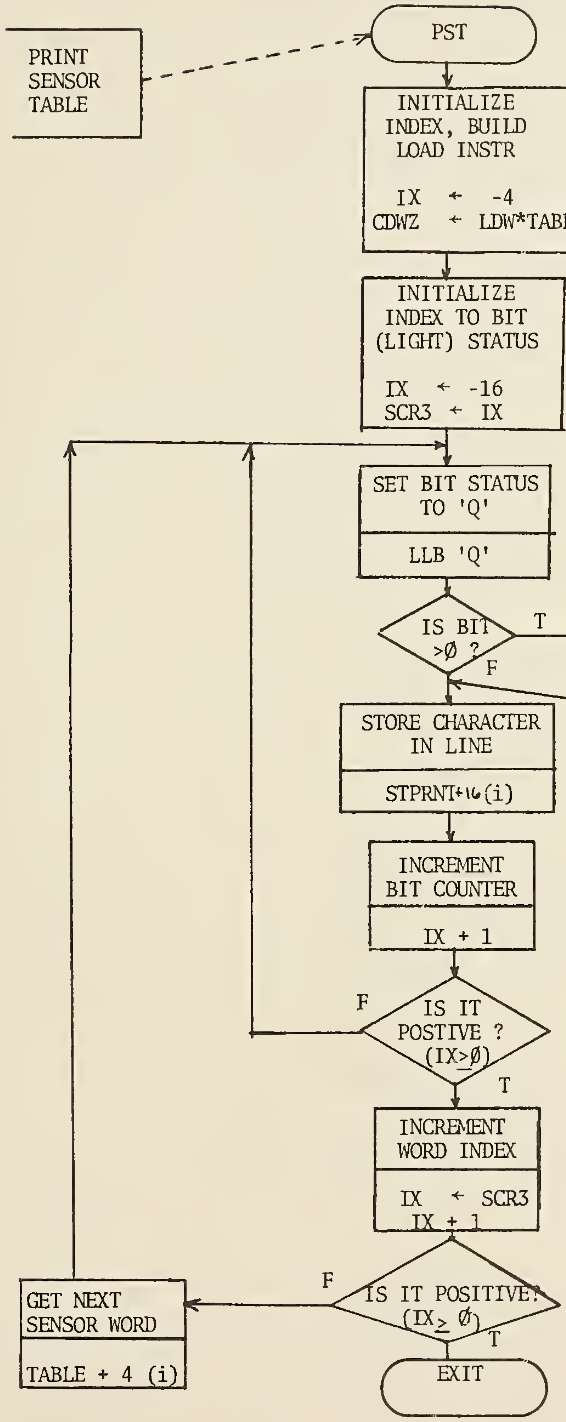

CJ
FAULT MON START - UP
INITIALIZE CLOCKS AND SENSOR TABLES
CCLK
CRNT
PCLK
NCNT
SWI(104)
← Ø
INITIALIZE SENSOR FAULT LISTS
SWO(24) ← X'FFFF'
CONTINUE AT PREIN
PRRG
CONVERT SEC COUNT
SECNT
CONVERT BINARY TO DECIMAL
CNVISP
CONVERT BCD TO ASCII
SDCHAR
REPEAT SEQUENCE FOR MINUTES
MINAD
A

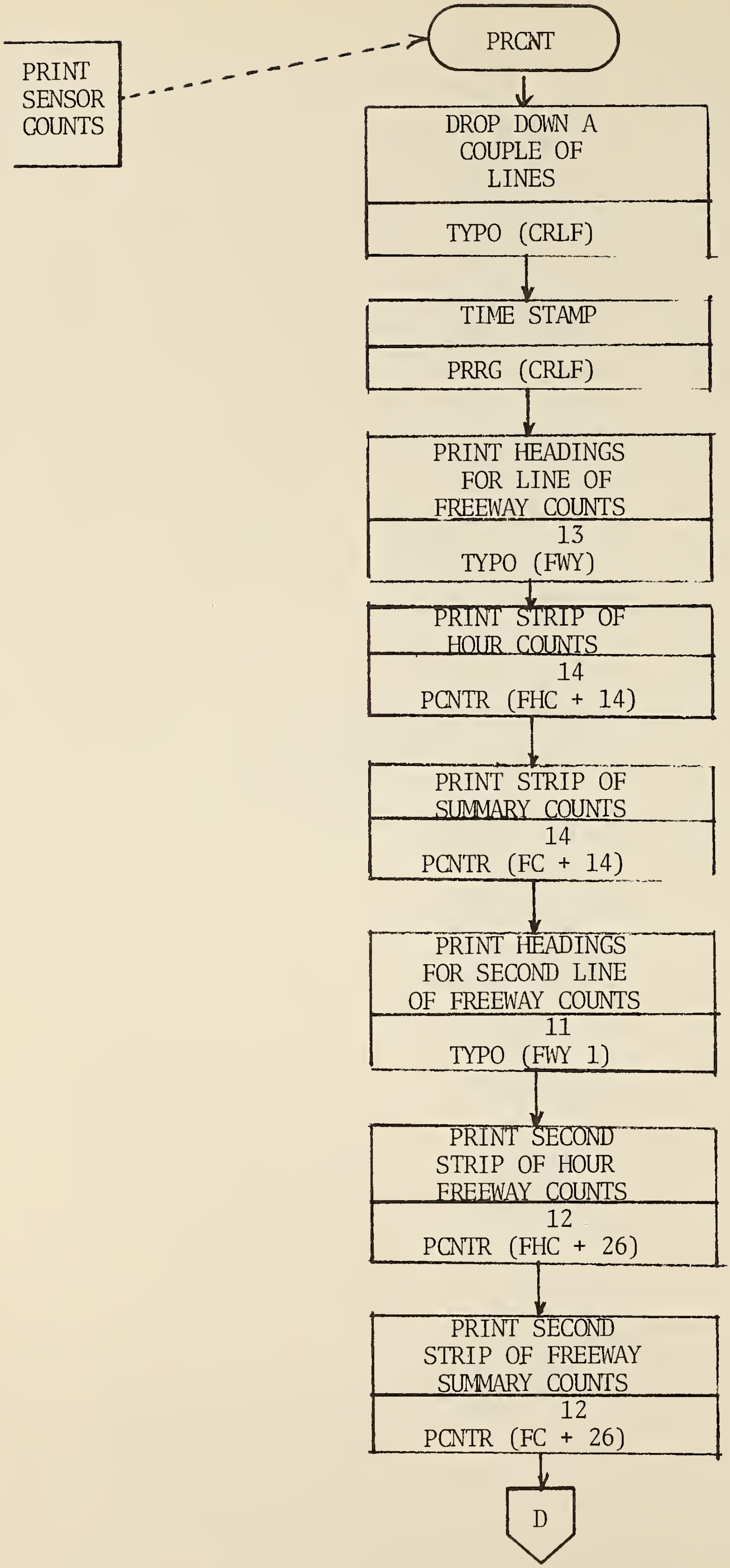

A
OUT PUT TIME STAMP
TYPO
EXIT TO TYPO FOR MSG
TYPO
STORE BREAK COUNT AND MESSAGE ADDRESS
BCNT
MSG
C
FORMER SET-UP CODE UNMODIFIED
ADDITION AT PNC
GET NEXT CHARACTER FROM MESSAGE
MSG(i)
BREAK CHARACTER?
F
T
B
DECREMENT BREAK COUNT
BCNT ← BCNT-1
BCNT NEGATIVE?
F
T
EXIT

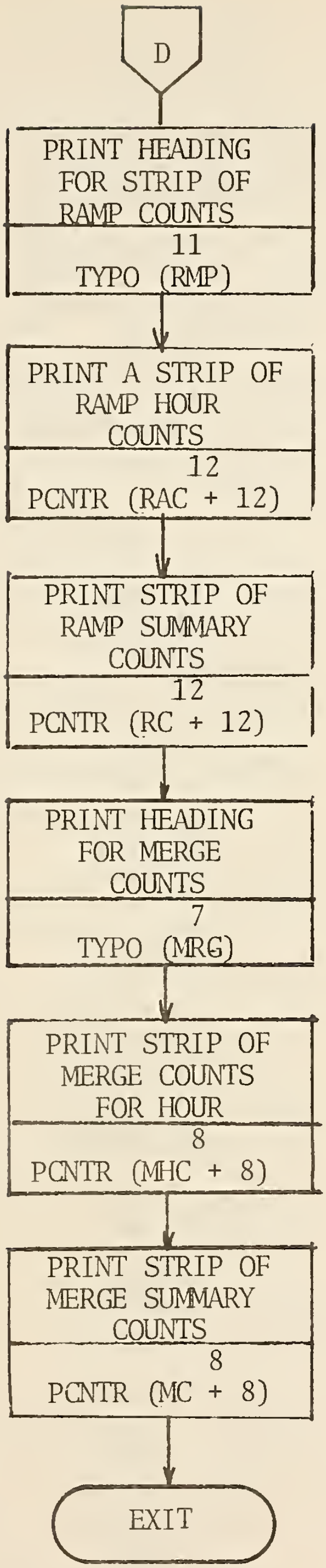

B
RETURN '/' CHARACTER?
T
F
CARRIAGE RETURN OUTPUT
OUTCHAR
OUTPUT MSG CHARACTER
OUTCHAR
CARRAIGE RETURN OUTPUT
OUTCHAR
LINE FEED OUTPUT
OUTCHAR
C

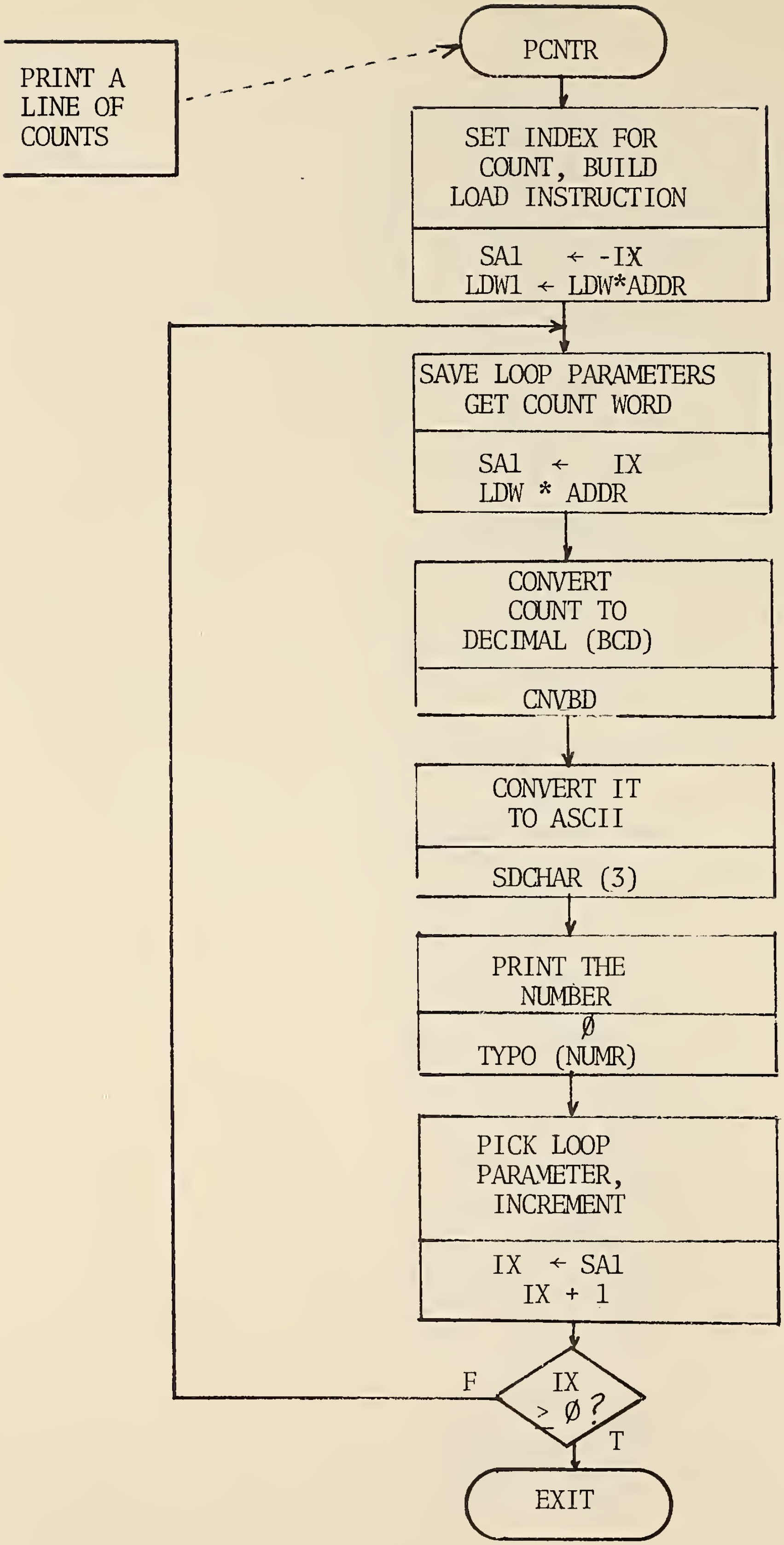

OUTCHAR
SET PRINT FLAG
FTRG ← CMRG
OUTPUT CHARACTER TO TELETYPE
CHECK ON PRINT STATUS
PCHK
F
IS IT DONE? (FTRG ≥ Ø)
T
EXIT

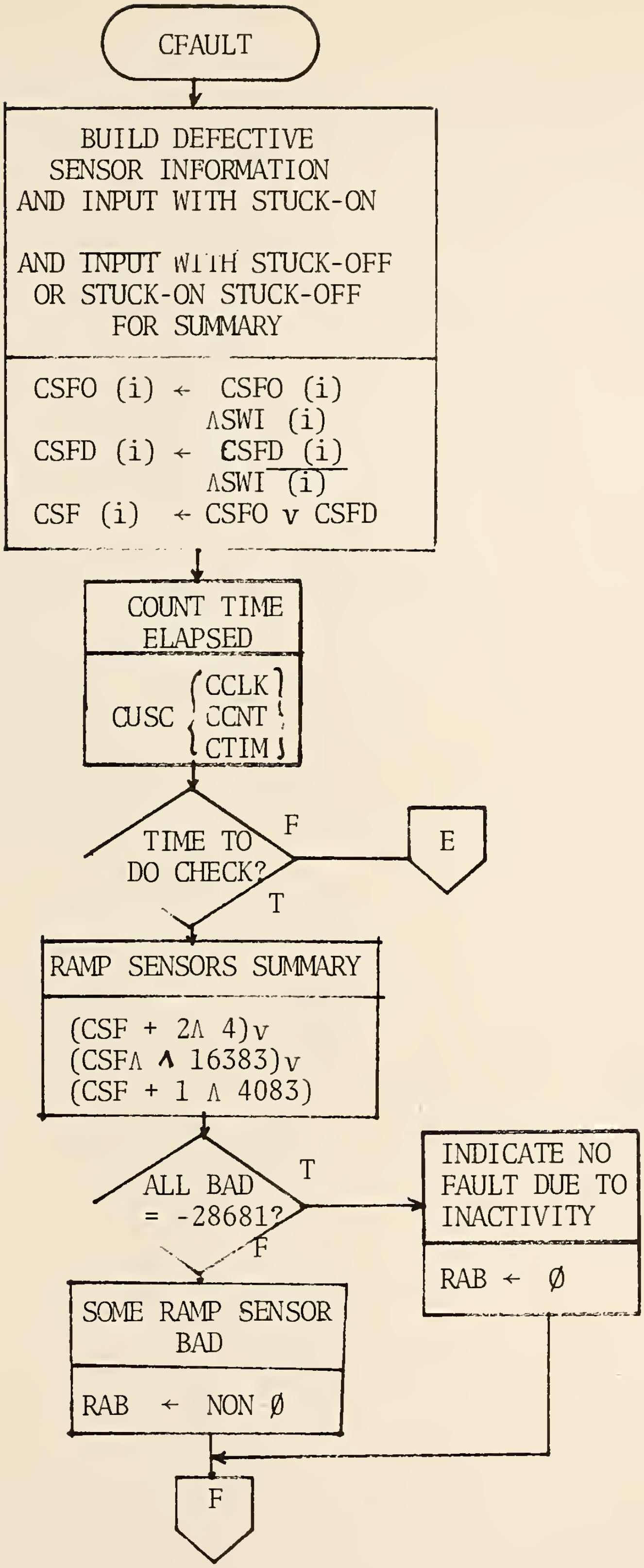

SMØ2
REPLACE ERROR CHECKS
INPUT SENSOR DATA UPDATE TABLES
SINP
CHECK FOR CRITICAL FAULTS
CFAULT
CHECK FOR NONCRITICAL FAULTS
NCFAULT
TYPE-OUT OF COUNTS REQUESTED? (CTFG ≠ Ø)
F
PRINT SENSOR COUNTS
PRCNT
SPØØ

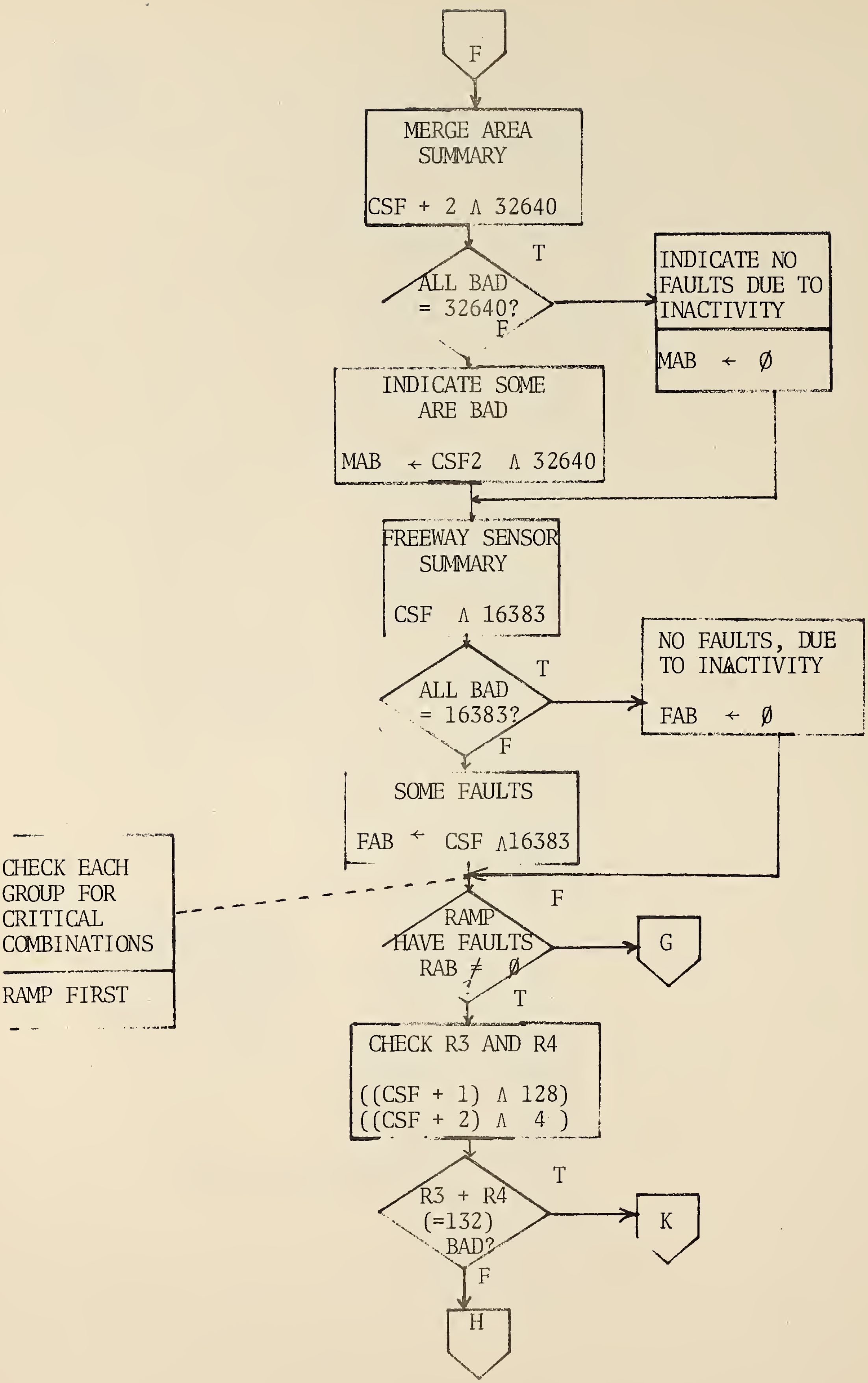

PRINT SENSOR TABLE
PST
INITIALIZE INDEX, BUILD LOAD INSTR
IX ← -4
CDWZ ← LDW*TABLE
INITIALIZE INDEX TO BIT (LIGHT) STATUS
IX ← -16
SCR3 ← IX
SET BIT STATUS TO 'Q'
LLB 'Q'
IS BIT >Ø ?
T
F
REPLACE IT WITH 'f'
LLB 'F'
STORE CHARACTER IN LINE
STPRNT+16(i)
INCREMENT BIT COUNTER
IX + 1
F
IS IT POSTIVE ? (IX≥Ø)
T
INCREMENT WORD INDEX
IX ← SCR3
IX + 1
F
IS IT POSITIVE? (IX≥ Ø)
T
GET NEXT SENSOR WORD
TABLE + 4 (i)
EXIT

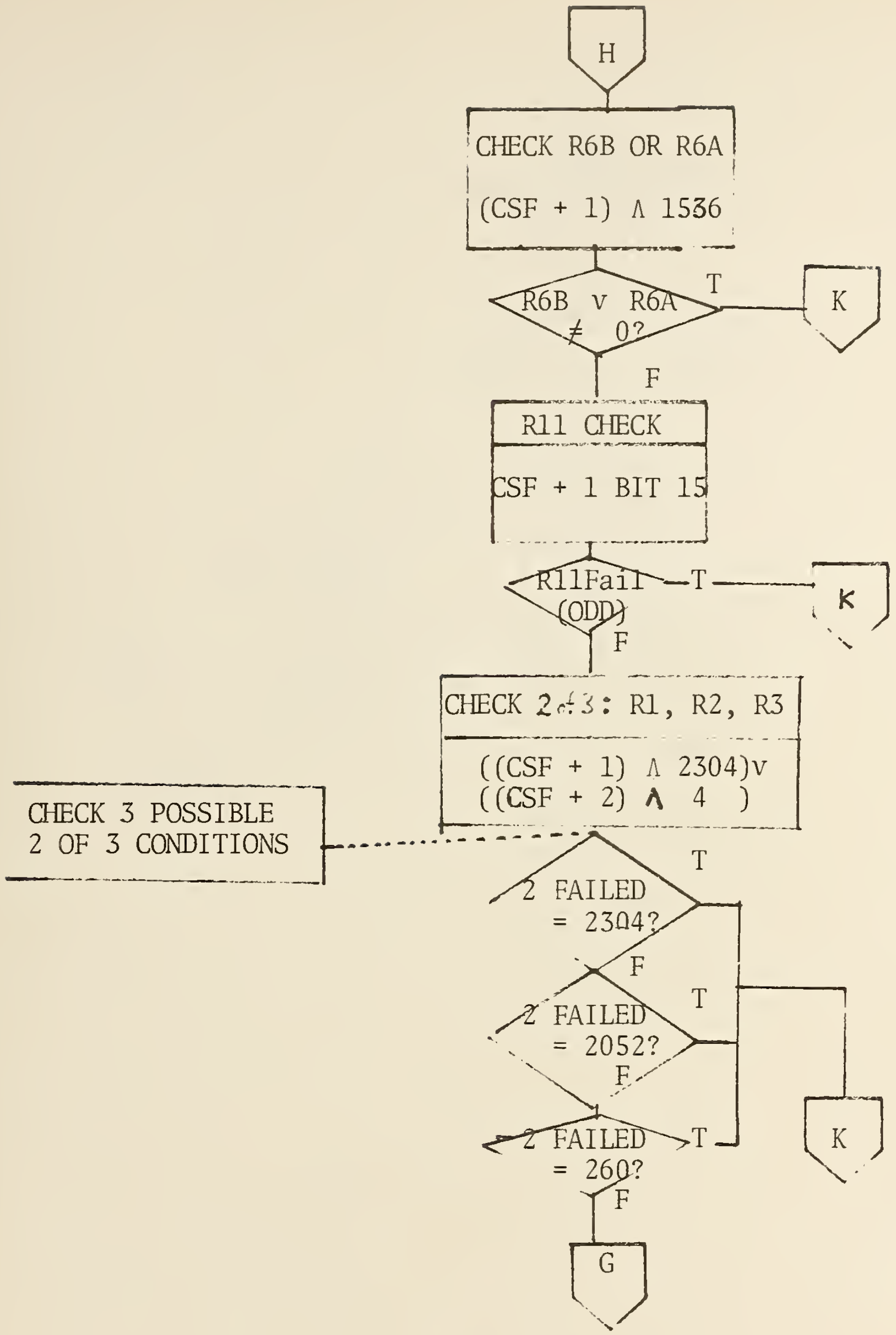

PRCNT
PRINT SENSOR COUNTS
DROP DOWN A COUPLE OF LINES
TYPO (CRLF)
TIME STAMP
PRRG (CRLF)
PRINT HEADINGS FOR LINE OF FREEWAY COUNTS
13
TYPO (FWY)
PRINT STRIP OF HOUR COUNTS
14
PCNTR (FHC + 14)
PRINT STRIP OF SUMMARY COUNTS
14
PCNTR (FC + 14)
PRINT HEADINGS FOR SECOND LINE OF FREEWAY COUNTS
11
TYPO (FWY 1)
PRINT SECOND STRIP OF HOUR FREEWAY COUNTS
12
PCNTR (FHC + 26)
PRINT SECOND STRIP OF FREEWAY SUMMARY COUNTS
12
PCNTR (FC + 26)
D

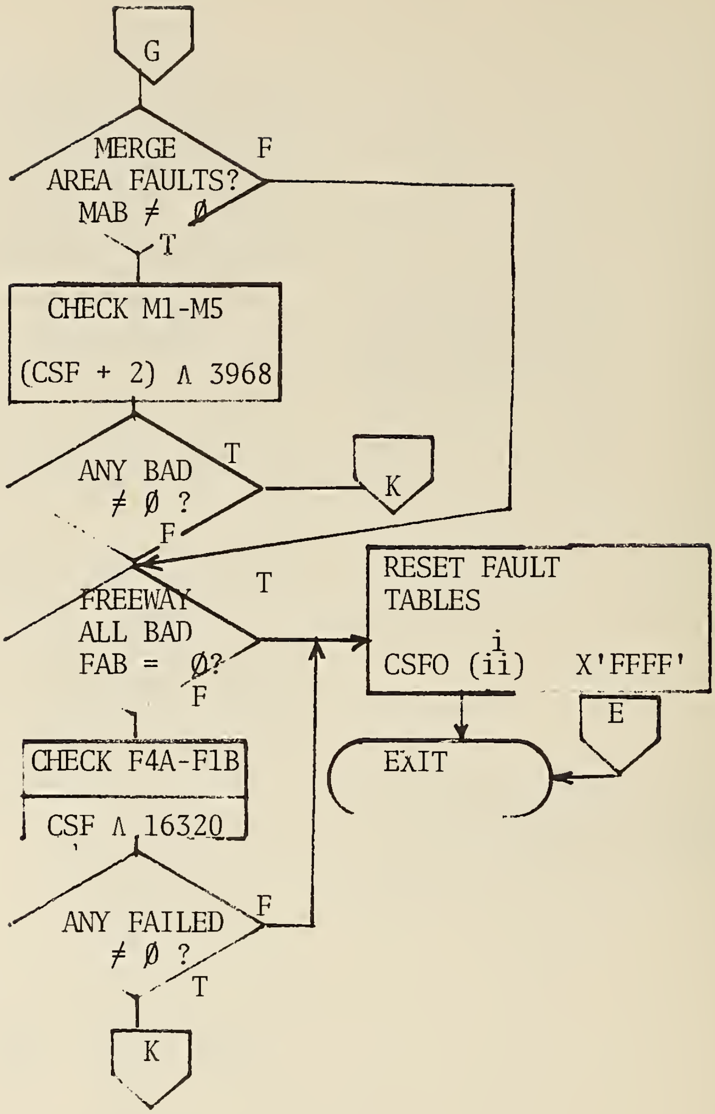

D
PRINT HEADING FOR STRIP OF RAMP COUNTS
11
TYPO (RMP)
PRINT A STRIP OF RAMP HOUR COUNTS
12
PCNTR (RAC + 12)
PRINT STRIP OF RAMP SUMMARY COUNTS
12
PCNTR (RC + 12)
PRINT HEADING FOR MERGE COUNTS
7
TYPO (MRG)
PRINT STRIP OF MERGE COUNTS FOR HOUR
8
PCNTR (MHC + 8)
PRINT STRIP OF MERGE SUMMARY COUNTS
8
PCNTR (MC + 8)
EXIT

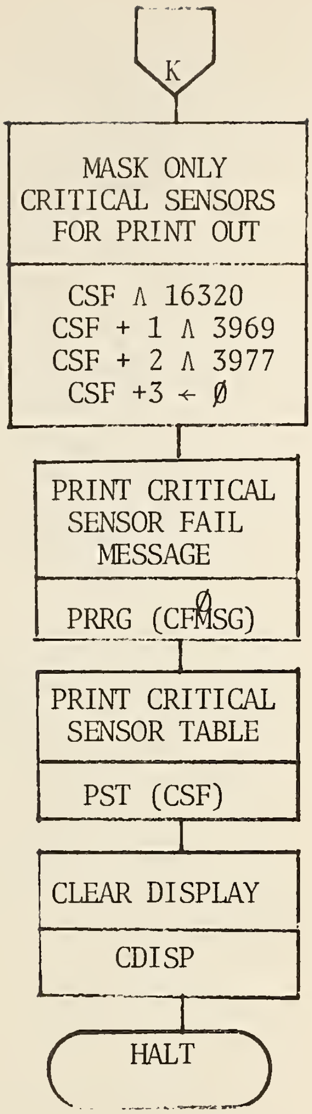

PRINT A LINE OF COUNTS
PCNTR
SET INDEX FOR COUNT, BUILD LOAD INSTRUCTION
SA1 ← -IX
LDW1 ← LDW*ADDR
SAVE LOOP PARAMETERS GET COUNT WORD
SA1 ← IX
LDW * ADDR
CONVERT COUNT TO DECIMAL (BCD)
CNVBD
CONVERT IT TO ASCII
SDCHAR (3)
PRINT THE NUMBER
TYPO (NUMR)
PICK LOOP PARAMETER, INCREMENT
IX ← SA1
IX + 1
F
IX ≥ Ø ?
T
EXIT

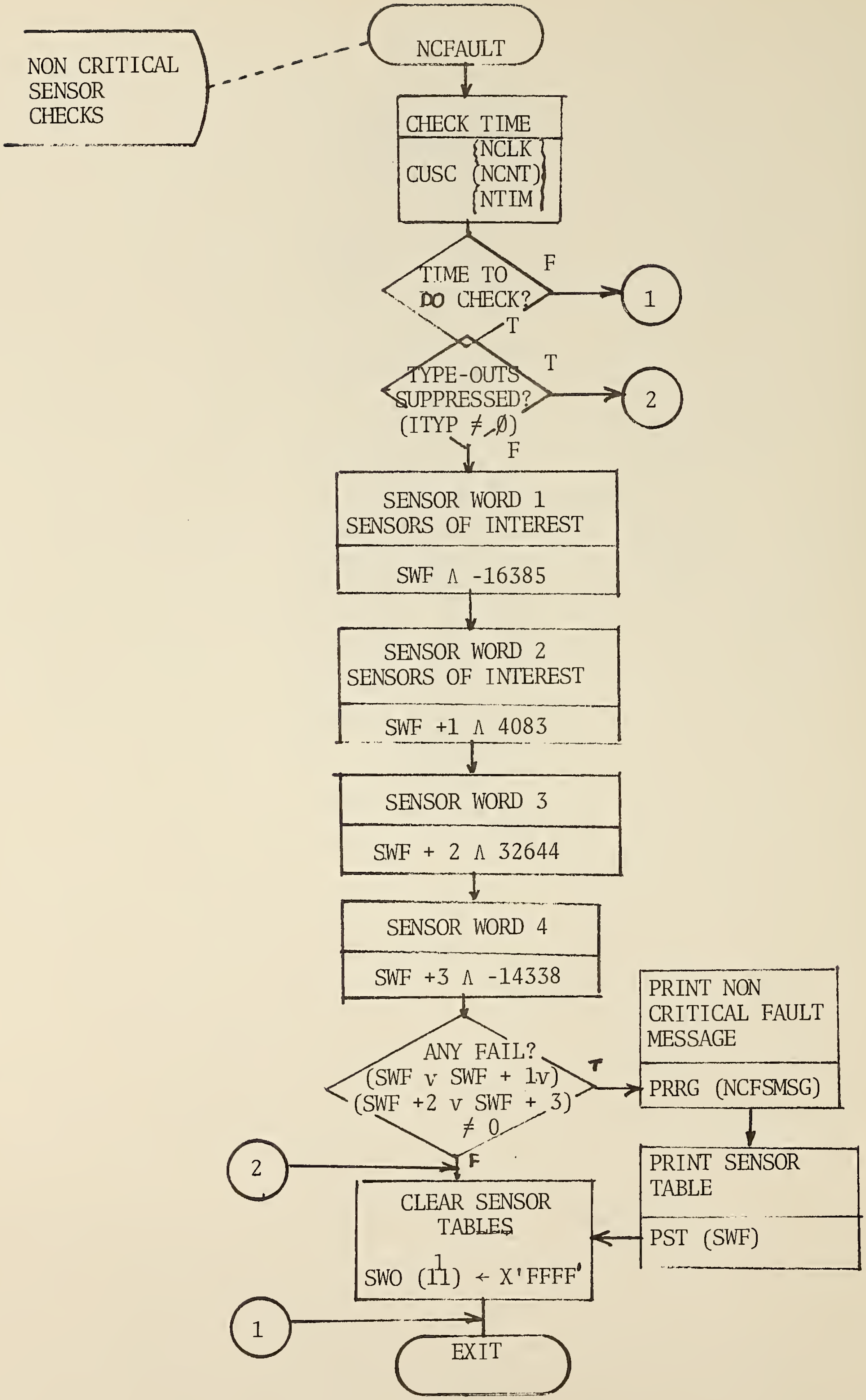

CFAULT
BUILD DEFECTIVE SENSOR INFORMATION AND INPUT WITH STUCK-ON
AND INPUT WITH STUCK-OFF OR STUCK-ON STUCK-OFF FOR SUMMARY
CSFO (i) ← CSFO (i) ∧SWI (i)
CSFD (i) ← CSFD (i) ∧SWI (i)
CSF (i) ← CSFO v CSFD
COUNT TIME ELAPSED
CUSC {CCLK CCNT CTIM}
TIME TO DO CHECK?
F
E
T
RAMP SENSORS SUMMARY
(CSF + 2∧ 4)v
(CSF∧ ∧ 16383)v
(CSF + 1 ∧ 4083)
ALL BAD = -28681?
T
INDICATE NO FAULT DUE TO INACTIVITY
RAB ← Ø
F
SOME RAMP SENSOR BAD
RAB ← NON Ø
F

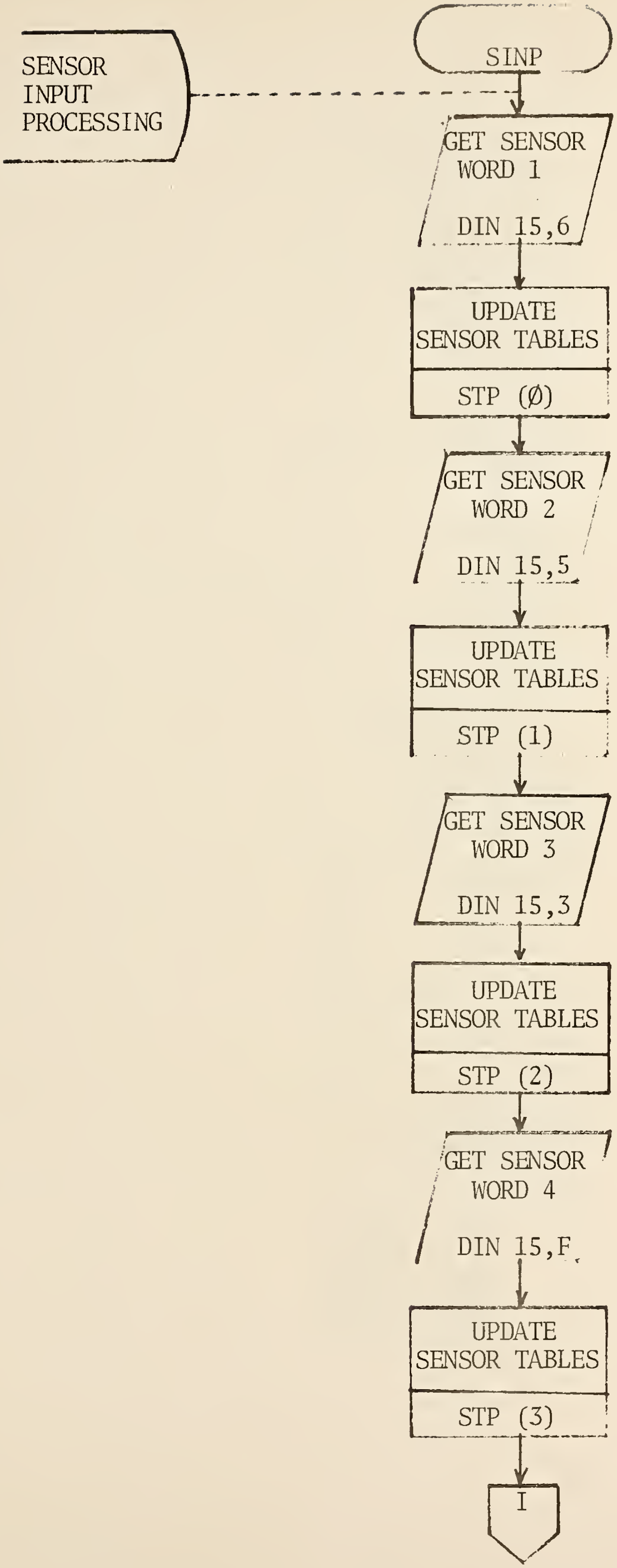

F
MERGE AREA SUMMARY
CSF + 2 ∧ 32640
ALL BAD = 32640?
T
F
INDICATE NO FAULTS DUE TO INACTIVITY
MAB ← Ø
INDICATE SOME ARE BAD
MAB ← CSF2 ∧ 32640
FREEWAY SENSOR SUMMARY
CSF ∧ 16383
ALL BAD = 16383?
T
F
NO FAULTS, DUE TO INACTIVITY
FAB ← Ø
SOME FAULTS
FAB ← CSF ∧16383
CHECK EACH GROUP FOR CRITICAL COMBINATIONS
RAMP FIRST
RAMP HAVE FAULTS RAB ≠ Ø
F
T
G
CHECK R3 AND R4
((CSF + 1) ∧ 128)
((CSF + 2) ∧ 4)
R3 + R4 (=132) BAD?
T
F
K
H

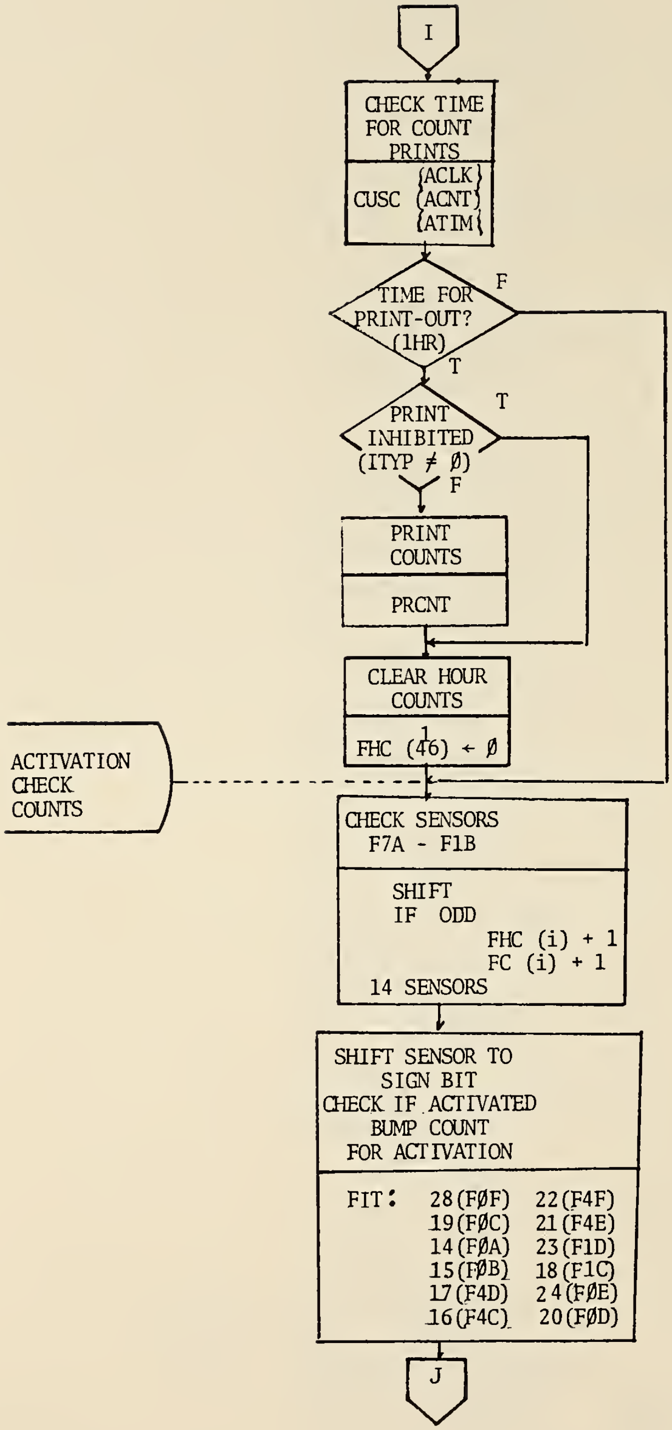

H
CHECK R6B OR R6A
(CSF + 1) Λ 1536
R6B v R6A ≠ 0?
T
K
F
R11 CHECK
CSF + 1 BIT 15
R11Fail (ODD)
T
K
F
CHECK 2 of 3: R1, R2, R3
((CSF + 1) Λ 2304)v
((CSF + 2) Λ 4)
CHECK 3 POSSIBLE
2 OF 3 CONDITIONS
2 FAILED = 2304?
T
F
2 FAILED = 2052?
T
F
2 FAILED = 260?
T
F
K
G

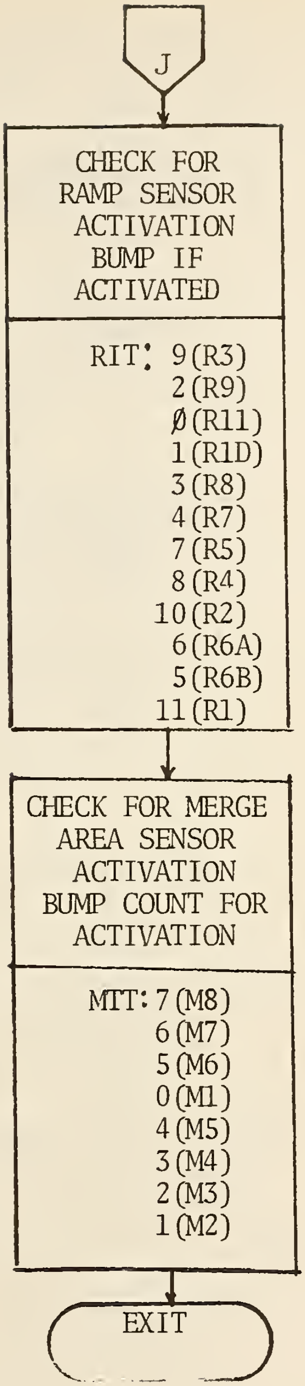

G
MERGE
AREA FAULTS?
MAB ≠ Ø
F
T
CHECK M1-M5
(CSF + 2) ∧ 3968
ANY BAD
≠ Ø ?
T
K
F
FREEWAY
ALL BAD
FAB = Ø?
T
F
RESET FAULT
TABLES
CSFO (ii) X'FFFF'
i
E
EXIT
CHECK F4A-F1B
CSF ∧ 16320
ANY FAILED
≠ Ø ?
F
T
K

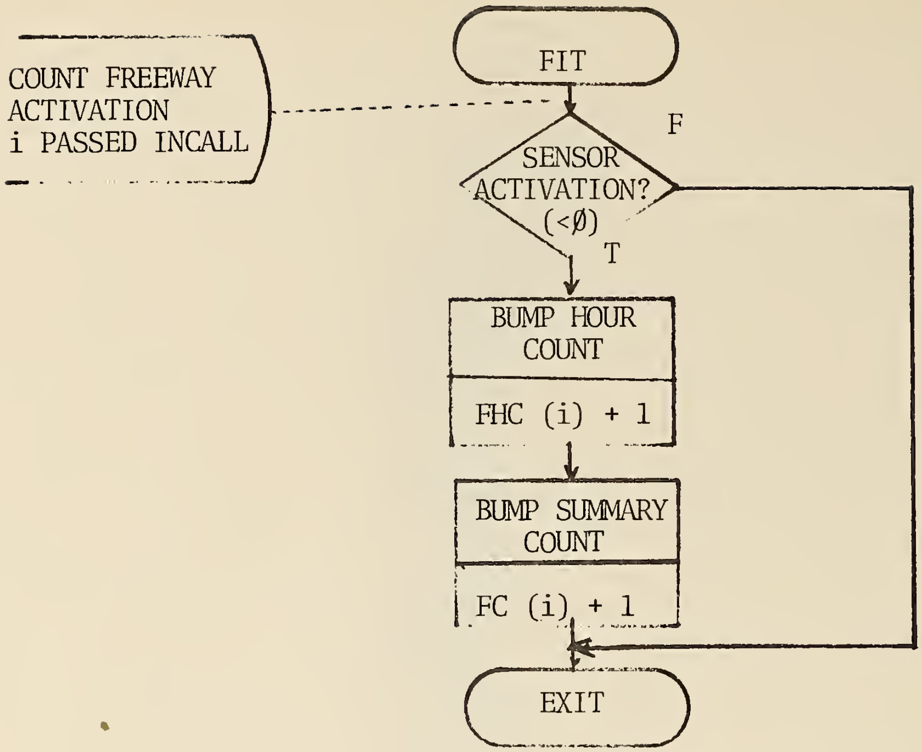

K
MASK ONLY CRITICAL SENSORS FOR PRINT OUT
CSF ∧ 16320
CSF + 1 ∧ 3969
CSF + 2 ∧ 3977
CSF +3 ← Ø
PRINT CRITICAL SENSOR FAIL MESSAGE
PRRG (CFMSG) Ø
PRINT CRITICAL SENSOR TABLE
PST (CSF)
CLEAR DISPLAY
CDISP
HALT


NON CRITICAL SENSOR CHECKS
NCFAULT
CHECK TIME
CUSC {NCLK (NCNT) NTIM}
TIME TO DO CHECK?
F
T
1
TYPE-OUTS SUPPRESSED? (ITYP ≠ Ø)
T
F
2
SENSOR WORD 1 SENSORS OF INTEREST
SWF ∧ -16385
SENSOR WORD 2 SENSORS OF INTEREST
SWF +1 ∧ 4083
SENSOR WORD 3
SWF + 2 ∧ 32644
SENSOR WORD 4
SWF +3 ∧ -14338
ANY FAIL? (SWF v SWF + 1v) (SWF +2 v SWF + 3) ≠ 0
T
F
PRINT NON CRITICAL FAULT MESSAGE
PRRG (NCFSMSG)
PRINT SENSOR TABLE
PST (SWF)
2
CLEAR SENSOR TABLES
SWO (11) ← X'FFFF'
1
EXIT

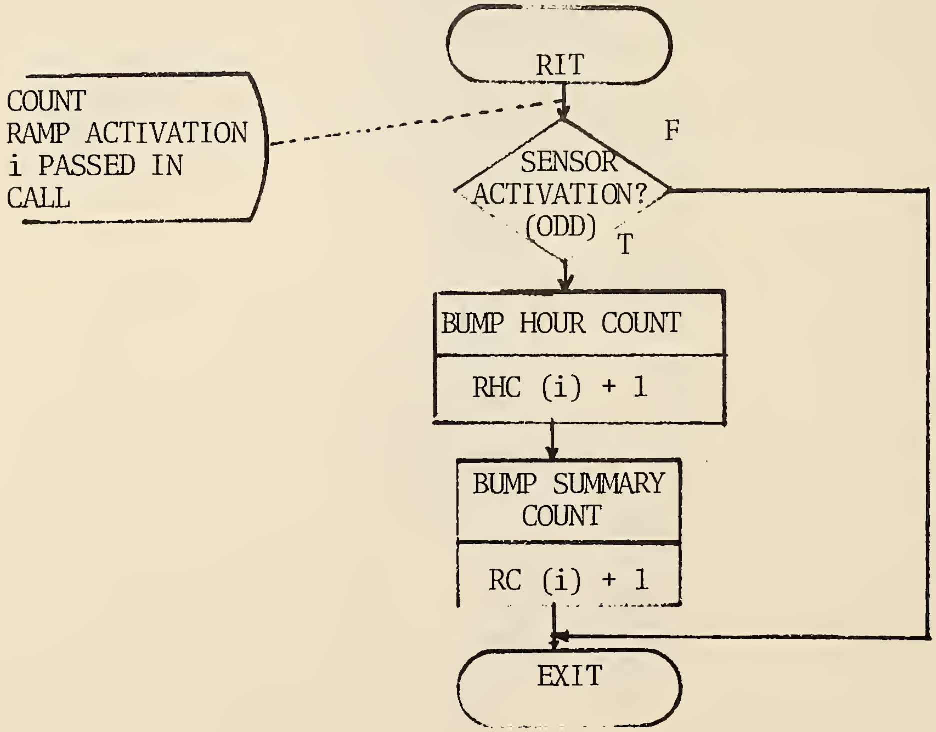

SENSOR
INPUT
PROCESSING
SINP
GET SENSOR
WORD 1
DIN 15,6
UPDATE
SENSOR TABLES
STP (∅)
GET SENSOR
WORD 2
DIN 15,5
UPDATE
SENSOR TABLES
STP (1)
GET SENSOR
WORD 3
DIN 15,3
UPDATE
SENSOR TABLES
STP (2)
GET SENSOR
WORD 4
DIN 15,F
UPDATE
SENSOR TABLES
STP (3)
I


I
CHECK TIME FOR COUNT PRINTS
CUSC {ACLK ACNT ATIM}
TIME FOR PRINT-OUT? (1HR)
F
T
PRINT INHIBITED (ITYP ≠ Ø)
T
F
PRINT COUNTS
PRCNT
CLEAR HOUR COUNTS
FHC (4$\overset{1}{}$6) ← Ø
ACTIVATION CHECK COUNTS
CHECK SENSORS F7A - F1B
SHIFT
IF ODD
FHC (i) + 1
FC (i) + 1
14 SENSORS
SHIFT SENSOR TO SIGN BIT
CHECK IF ACTIVATED
BUMP COUNT FOR ACTIVATION
FIT: 28(FØF) 22(F4F)
19(FØC) 21(F4E)
14(FØA) 23(F1D)
15(FØB) 18(F1C)
17(F4D) 24(FØE)
16(F4C) 20(FØD)
J

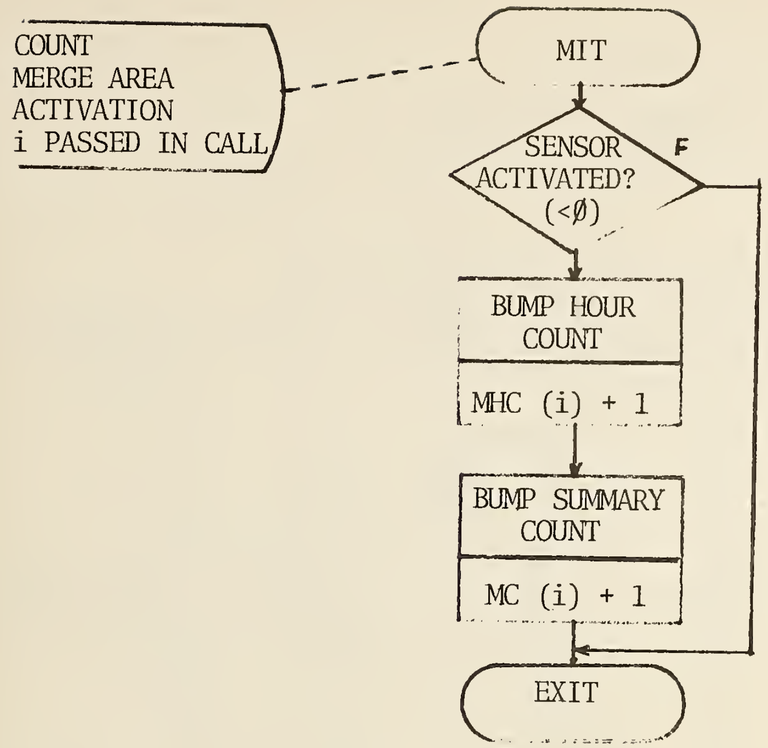

J
CHECK FOR RAMP SENSOR ACTIVATION BUMP IF ACTIVATED
RIT: 9(R3)
2(R9)
0(R11)
1(R1D)
3(R8)
4(R7)
7(R5)
8(R4)
10(R2)
6(R6A)
5(R6B)
11(R1)
CHECK FOR MERGE AREA SENSOR ACTIVATION BUMP COUNT FOR ACTIVATION
MTT: 7(M8)
6(M7)
5(M6)
0(M1)
4(M5)
3(M4)
2(M3)
1(M2)
EXIT

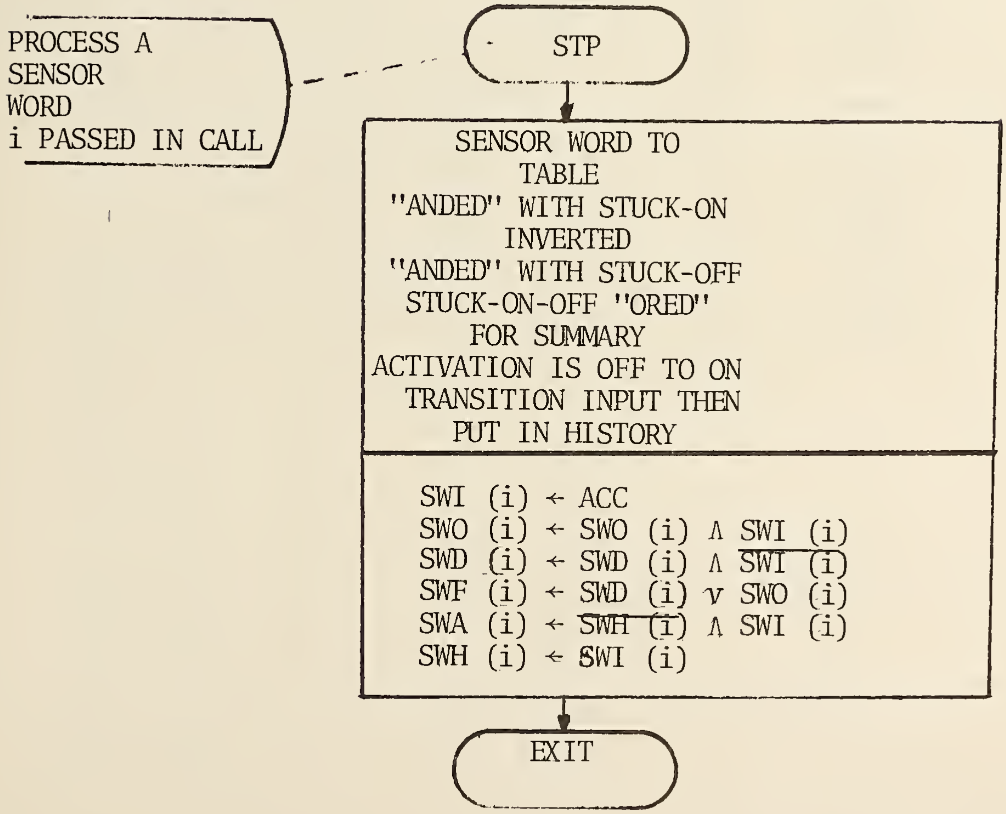


FIT
COUNT FREEWAY ACTIVATION i PASSED INCALL
SENSOR ACTIVATION? (<Ø)
F
T
BUMP HOUR COUNT
FHC (i) + 1
BUMP SUMMARY COUNT
FC (i) + 1
EXIT



RIT
COUNT RAMP ACTIVATION i PASSED IN CALL
SENSOR ACTIVATION? (ODD)
F
T
BUMP HOUR COUNT
RHC (i) + 1
BUMP SUMMARY COUNT
RC (i) + 1
EXIT


COUNT
MERGE AREA
ACTIVATION
i PASSED IN CALL
MIT
SENSOR
ACTIVATED?
(<Ø)
F
BUMP HOUR
COUNT
MHC (i) + 1
BUMP SUMMARY
COUNT
MC (i) + 1
EXIT


PROCESS A
SENSOR
WORD
i PASSED IN CALL
STP
SENSOR WORD TO
TABLE
"ANDED" WITH STUCK-ON
INVERTED
"ANDED" WITH STUCK-OFF
STUCK-ON-OFF "ORED"
FOR SUMMARY
ACTIVATION IS OFF TO ON
TRANSITION INPUT THEN
PUT IN HISTORY
SWI (i) ← ACC
SWO (i) ← SWO (i) ∧ SWI (i)
SWD (i) ← SWD (i) ∧ $\overline{\text{SWI (i)}}$
SWF (i) ← SWD (i) v SWO (i)
SWA (i) ← $\overline{\text{SWH (i)}}$ ∧ SWI (i)
SWH (i) ← SWI (i)
EXIT

```
              ' FAULT MONITOR SUBPROGRAM  #1
              * VERSION PREPARED 9-28-74
              *  STD. OF EXCEL. IS AVG OF F1 THRU F4
              *
              * VERSION OF 1/16/77
              * DOES NOT USE HVP IN FAULT CHECKS
              * HAS CRITICAL SENSOR CHECKS
              * CAN SUPPRESS ALL BUT CRITICAL SENSOR MESSAGES
              * CAN PRINT SENSOR COUNTS FOR HOUR AND SUMMARY
              *
                         ORIG   X'2800'
2800 100F     SARG       JMP    STAG   INITIALIZATION ENTRY
2801 10EC     SERG       JMP    ERG   NORMAL ENTRY
     0021     TIME       EQU    X'0021'    ' TIME REGISTER
     0020     BEGT       EQU    X'0020'    ' REINITIALIZATION LOC
     0046     ERRCH      EQU    X'0046'    ' WHO DONE IT CHARACTER
     062C     TRAG       EQU    X'062C'    ' OPERATOR MONITOR SUBP
     0B88     FTTY       EQU    X'0B88'    ' MISS T1 HIGHW
     0B8F     FIRY       EQU    X'0B8F'    ' MISS T3 HIGHW
     0BAB     FNVY       EQU    X'0BAB'    ' NEW CAR HIGHW
     1802     MODE       EQU    X'1802'    ' SYSTEM MODE
     180C     K51F       EQU    X'180C'    ' BAND MASKING WORD
     1617     TYPEFLG    EQU    X'1617'    'DATA LOG INITIAL MSG
     0027     RERG       EQU    X'0027'    'READ ADDRESS OPER MONITOR
     3EE8     CUSC       EQU    X'3EE8' CHECK SECOND COUNT ROUTINE
2802 0000     TMREQ      DATA   0  ' TIME PRINT REQUEST FLAG
2803 0001     ITYP       D      1  TYPEOUTS INHIBITED AT LOAD
2804 0000     CTFG       D      0  PRINT SENSOR COUNTS OFF
2805 000A     CHLG       D      10    H LOOP CONSTANT
2806 0014     CHSG       D      20    H SENSOR CONSTANT
2807 000A     CRLG       D      10    R LOOP CONSTANT
2808 0014     CRSG       D      20    R SENSOR CONSTANT
2809 0905     PERG       D      X'0905'   TRAFFIC PERCENTAGE
280A 100B     SORG       JMP    STOP   GO TO HALT TRAP
```

```
                'FAULT MONITOR SUBPROGRAM  #2
280B 0000       STOP      HLT   ***** HALT TRAP *****
280C 100B                 JMP   STOP
                * FAULT MON START UP SUBROUTINE
280D 0000                 D     0
280E 28A6                 D     IRRG   HARDW IRPT ADDRESS
280F 600D       STAG      STX   $-2
2810 0100                 CLR
2811 73B2                 STW   FERG
2812 7002                 STW   TMREQ
2813 95D2                 LDX   =103
2814 0040                 SLM
     2815       CJ        EQU   $
2815 7B1D                 STW * SWI    FAULT TABLES AND COUNTS
2816 0501                 DXS   1
2817 1015                 JMP   CJ
2818 7460                 STW   CCLK
2819 7461                 STW   CCNT
281A 74D0                 STW   NCLK
281B 74D1                 STW   NCNT
281C 7004                 STW   CTFG
                *
281D 0120                 INV    INIT -1, ALL SET
281E 95D3                 LDX   =23
     281F       SJ        EQU   $
281F 7B05                 STW * SWO  ALL SENSOR FAULT TABLES
2820 0501                 DXS   1
2821 101F                 JMP   SJ
                *
2822 03C0                 SS0
2823 1039                 JMP   PLDM2
2824 20DB                 JSX   CDISP
2825 0080                 SMB   TIME   INIT PRINTING LOGIC
2826 8021                 LDW   TIME
2827 0080                 SMB   TIME
2828 C022                 ORI   TIME+1
2829 0800                 SAZ
282A 1033                 JMP   PREIN   AT ZERO TIME PRINT
282B 0100                 CLR
282C 739E                 STW   RNTIME  "FULL INIT"
282D 83B1                 LDW   DERG
282E 73C5                 STW   TIRG
282F 0100                 CLR
2830 205B                 JSX   PRRG
2831 54BC                 D     /MERG
2832 1039                 JMP   PLDM2
2833 0080       PREIN     SMB   ERRCH   IF NOT ZERO TIME THEN
2834 8046                 LDW   ERRCH
2835 7258                 STW   MDBG    GET WHO-DONE-IT CHARACTER
2836 0100                 CLR    AND PRINT REINIT MESSAGE
2837 205B                 JSX   PRRG
2838 54B0                 D     /MDRG
2839 83A9       PLDM2     LDW   CM2G
283A 73B4                 STW   FSTG   SET START UP F
283B 719C                 STW   FMODE
283C 719D                 STW   FBMSK
283D 800E                 LDW   STAG-1
```

```
283E 0080     93           SML   0
283F 7019     94           STW   25
2840 034F     95           DOT   4,X'F'   RESET FAULT MON HARDW
2841 0036     96           DSB   6
2842 80C2     97           LDW   PFRTN-1   PWR FAIL SERVICE ADDR
2843 0080     98           SML   0
2844 701D     99           STW   29
2845 0037    100           DSB   7
2846 0027    101           ENB   7
2847 900D    102           LDX   STAG-2
2848 2800    103           JSX * 0
2849 0000    104           D     0    *** PRINT SERVICE ROUTINE ***
284A 0100    105 PTRG      CLR
284B 73B5    106           STW   FTRG   CLEAR PRINT F
284C 03E0    107           DOT   X'E',0   DESELECT TTY
284D 2800    108           JSX * 0
             109 *
             110 *STATUS CHECK SUBROUTINE
             111 *
284E 0000
284F 604E    112 PCHK      SUBR
2850 02E0    113           DIN   X'E',0
2851 0AC1    114           SRC L 1
2852 0820    115           SAM
2853 1057    116           JMP   PCDN
2854 0080    117           SML   0
2855 9001    118           LDX   1
2856 2800    119           JSX * 0
2857 904E    120 PCDN      EXIT  PCHK
2858 2800
2859 0000    121           D     0   *** PRINT SUBROUTINE ***
             122 *
             123 * TYPE-OUTS
             124 *    CALL:
             125 *           PRRG (TIME STAMPED TYPE-OUTS)
             126 *  -OR-     TYPO (NO TIME ON TYPE-OUT)
             127 *(0)       D     BYTE ADDRESS TEXT
             128 *(1)       RETURN
             129 *   ACCUMULATOR - NUMBER ':' BYTES TO PASS
             130 *                              (NOT TYPED)
             131 * TYPE-OUT WITH TIME STAMP
             132 *
285A 0000
285B 605A    133 PRRG      SUBR
285C 7387    134           STW   PS1  PRESERVE MSG VALUE
285D 21CD    135           JSX   SECNT
285E 219F    136           JSX   CNVBD
285F 21E8    137           JSX   SDCHAR  STORE SEC CHARACTERS
2860 0001    138           D     1
2861 542E    139           D     /SECLC
2862 839E    140           LDW   RNTIME
2863 A1E6    141           ADD   MINADD
2864 219F    142           JSX   CNVBD
2865 21E8    143           JSX   SDCHAR
2866 0002    144           D     2
2867 542A    145           D    /MINLC
2868 0100    146           CLR
```

```
2869 206F             JSX   TYPO   WHAT TIME
286A 542A             D     /TMSTMP
286B 905A             LDX   PRRG-1
286C 8387             LDW   PSI
286D 106F             JMP   TYPO
                *
                * TYPE-OUT WITHOUT TIME STAMP
                *
286E 0000
286F 606E       TYPO      SUBR
2870 7386             STW   BCNT  BREAK COUNT
2871 8800             LDW * 0
2872 7095             STW   MSG   MESSAGE ADDRESS
                *
2873 83B2             LDW   FERG
2874 0810             SAP     PREVIOUS FAULT 0
2875 107C             JMP   PR00    YES
2876 83C3             LDW   PNRG   PTRG ADDRESS
2877 0080             SML   0
2878 7001             STW   1   PRINT LINKAGE ADDRESS
2879 83A9             LDW   CM2G
287A 73B2             STW   FERG  SET PREV FAULT F
287B 73B7             STW   FWAG  SET RETYPE FLAG
     287C       PR00      EQU   $
                *
     287C       PNC       EQU   $
287C 9095             LDX   MSG  PICK UP MSG CHAR
287D 5800             LDB * 0
                *
287E 07BA             CLB ':'  BREAK CHECK
287F 0360             SEQ
2880 1088             JMP   T1  IT WAS NOT
2881 9386             LDX   BCNT  GET COUNT
2882 0501             DXS   1  DECREMENT
2883 1086             JMP   NOTND  WAS NOT END
2884 906E             EXIT  TYPO,1  WAS END, GO
2885 2801
2886 6386       NOTND     STX   BCNT
2887 1093             JMP   NXTCH
                *
     2888       T1        EQU   $
2888 07AF             CLB   '/'  WAS IT RETURN
2889 0860             SEQ
288A 1092             JMP   T2  NO, PUT CHAR OUT
288B 068D             LLB   141
288C 2099             JSX   OUTCHAR  YES, CR, CR, LF
288D 068D             LLB   141
288E 2099             JSX   OUTCHAR
288F 068A             LLB   138
2890 2099             JSX   OUTCHAR
2891 1093             JMP   NXTCH
                *
     2892       T2        EQU   $
2892 2099             JSX   OUTCHAR
     2893       NXTCH     EQU   $
2893 9095             LDX   MSG   NEXT MESSAGE CHAR
2894 0401             IXS   1
```

```
     FAULT MONITOR SUBPROGRAM  #2                           PAGE  5
2895            MSG       RES   1
2896 6095                 STX   MSG
2897 107C                 JMP   PNC


     FAULT MONITOR  #3                                      PAGE  6

                'FAULT MONITOR  #3
                *
2898 0000
2899 6098       OUTCHAR   SUBR
289A 0130                 CAX
289B 83A9                 LDW   CM2G
289C 73B5                 STW   FTRG
289D 0140                 CXA
289E 03EE                 DOT   14,14
     289F       TWAIT     EQU   $
289F 204F                 JSX   PCHK
28A0 83B5                 LDW   FTRG
28A1 0310                 SAP
28A2 109F                 JMP   TWAIT
28A3 9098                 EXIT  OUTCHAR
28A4 2800


      FAULT MONITOR  #4                                     PAGE  7

                ' FAULT MONITOR  #4
                * FAULT MON HARDWARE INTERRUPT SUBROUTINE
28A5 0000                 D     0
28A6 00A0       IRRG      MSK
28A7 0034                 DSB   4
28A8 0036                 DSB   6
28A9 70A5                 STW   IRRG-1
28AA 0100                 CLR
28AB 73B2                 STW   FERG   CLEAR PREV FAULT F
28AC 024F                 DIN   4,X'F'   INPUT F MON WORD
28AD 0830                 SAO      CLOCK OK 0
28AE 10BA                 JMP   IE01   NO
28AF 83B2                 LDW   FERG
28B0 0810                 SAP      FIND ANY FAULTS 0
28B1 10BD                 JMP   IE06   YES,STOP 704
28B2 0024                 ENB   4
28B3 83A9                 LDW   CM2G
28B4 73B6                 STW   FUNG   SET PRINT FAULT UNKM F
28B5 030F                 DOT   0,X'F'   PREVENT FLASHING LIGHTS
28B6 034F                 DOT   4,X'F'   RESET FAULT MON HARDW
28B7 80A5                 LDW   IRRG-1   RESTORE ACR
28B8 00B0                 UNM
28B9 0016                 INR   6
     28BA       IE01      EQU   $
28BA 0100                 CLR
28BB 205B                 JSX   PRRG
28BC 5432                 D     /MIRG   CLOCK OUT MSG
28BD 83C2       IE06      LDW   LKRG
28BE 0080                 SML   0
28BF 7001                 STW   1   LINKAGE ADDRESS
28C0 03EB                 DOT   X'E',X'B' ENABLT TTY KEYBOARD
28C1 200B                 JSX   STOP  GOTO HALT TRAP
```

```
          ' FAULT MONITOR  #5
          * POWER FAIL INTERRUPT HANDLER AND RESTART
28C2 28C3                  DATA  PFRTN
28C3 00A0   PFRTN          MSK
28C4 0100                  CLR
28C5 0390                  DOT   9,0
28C6 03E0                  DOT   X'E',0  RESET DEVICES
28C7 20DB                  JSX   CDISP  CLEAR FIELD
28C8 80CF                  LDW   PURTN-2
28C9 0080                  SML   0
28CA 7000                  STW   0      SET UP FOR RESTART
28CB 80D0                  LDW   PURTN-1
28CC 0080                  SML   0
28CD 7001                  STW   1
28CE 100B                  JMP   STOP
28CF 008A                  SMB   $   START UP ROUTINE
28D0 10D1                  JMP   PURTN
28D1 20DB   PURTN          JSX   CDISP   CLEAR DISPLAY
28D2 0100                  CLR
28D3 205B                  JSX   PRRG
28D4 547A                  D     /M7RG
28D5 06D0                  LLB   X'D0'   P IS HE WHO-DONE-IT
28D6 0080                  SMB   ERRCH
28D7 7046                  STW   ERRCH
28D8 0080                  SMB   BEGT    REINIT
28D9 2020                  JSX   BEGT
28DA 0000
28DB 60DA   CDISP          SUBR
28DC 0100                  CLR     SEND OUT ZERO TO FIRST
                           DO    1,10  10 WORDS OF FIELD
28DD 0301                  DOT   0,?
28DE 0302
28DF 0303
28E0 0304
28E1 0305
28E2 0306
28E3 0307
28E4 0308
28E5 0309
28E6 030A
28E7 83A6                  LDW   CWRD11  SEND LAST WORD WITH BITS
28E8 030B                  DOT   0,11    TO TURN ON YIELD AND AMBER
28E9 90DA                  EXIT  CDISP
28EA 2800
```

```
          ' FAULT MONITOR  #6
28EB 0000               DATA   0  CHECK SENSOR ERROR LISTS
28EC 0040      ERG      SLM       SUBPROGRAM ENTRY
28ED 83B4               LDW    FSTG
28EE 0810               SAP       START UP 0
28EF 1135               JMP    SM00   YES
28F0 83A2               LDW    CCLG  ADDR CARRY MEMORY TEST
28F1 0120               INV
28F2 A3A4               ADD    CFFG
28F3 F3A5               CMW    CFEG
28F4 0860               SEQ       704 ARITHMETIC OK 0
28F5 1159               JMP    SM01   NO
          * TIME AND HIGHW FAULT TOLERANCES
28F6 0100               CLR
28F7 73B2               STW    FERG   CLEAR PREV FAULT F
28F8 0080               SMB    TIME
28F9 8021               LDW    TIME
28FA B3C5               SUB    TIRG
28FB 0810               SAP       MORE THAN 1 MIN 0
28FC 1116               JMP    SM02   NO
28FD A3C5               ADD    TIRG   TIME RESTORED
28FE A3B1               ADD    DERG   DELTA TIME
28FF 73C5               STW    TIRG   NEXT 1 MIN
2900 839D               LDW    BCRG   1 MIN COUNTER
2901 A3AB               ADD    CIRG   ADD 1
2902 739D               STW    BCRG   INCREMENTED
2903 839E               LDW    RNTIME
2904 A3AB               ADD    CIRG  BUMP RUNNING TIME
2905 739E               STW    RNTIME
2906 83B7               LDW    FWAG
2907 0820               SAM       WAIT FOR RETYPE
2908 1116               JMP    SM02   NO
2909 839F               LDW    BC5G
290A A3AB               ADD    CIRG
290B 739F               STW    BC5G   10 MIN COUNTER
290C 070A               CLB    10
290D 0860               SEQ       COUNTER = 10 ?
290E 1116               JMP    SM02   NO
290F 0100               CLR       YES
2910 739F               STW    BC5G   RESET
2911 73B7               STW    FWAG   CLEAR RETYPE F
2912 93B0               LDX    C9RG   CLEAR RETYPE WORDS
2913 7BB8               STW  * HWA1
2914 0501               DXS    1
2915 1113               JMP    $-2
```

```
           * FAULT MONITOR  #7
     2916  SM02      EQU   $
2916 24FB            JSX   SINP
           *
2917 2450            JSX   CFAULT
           *
2918 24CD            JSX   NCFAULT
           *
2919 8004            LDW   CTFG  SENSOR COUNT REQUEST
291A 0800            SAZ
291B 111D            JMP   $+2
291C 1120            JMP   NCTOUT   SKIP IF NOT WANTED
291D 0100            CLR
291E 7004            STW   CTFG
291F 23E6            JSX   PRCNT
     2920  NCTOUT    EQU   $
```

```
                ' FAULT MONITOR  #8
2920 83B6  SP00      LDW   FUNG
2921 0820            SAM      PRINT FAULT UNKN 0
2922 1128            JMP   SM50   NO
2923 0100            CLR      YES
2924 73B6            STW   FUNG
2925 0100            CLR
2926 205B            JSX   PRRG
2927 546E            D     /M6RG
2928 115B  SM50      JMP   UF01
2929 0820            SAM      FIND ANY FAULTS 0
292A 112F            JMP   SM51   NO
292B 83C2            LDW   LKRG   YES
292C 0080            SML   0
292D 7001            STW   1   READ LINKAGE ADDRESS
292E 03EB            DOT   X'E',X'B'   SELECT KEYBOARD
292F 839D  SM51      LDW   BCRG
2930 B3C4            SUB   RESG
2931 0820            SAM
2932 1143            JMP   SM52   CLEAR HVP FAULT TABLES ETC.
2933 0081  SM53      SMB   TRAG
2934 162C            JMP   TRAG   GOTO OP. MONITOR ENTRY
2935 0100  SM00      CLR
2936 739F            STW   BC5G   ZERO 10 MIN COUNTER
2937 73B4            STW   FSTG   CLEAR START-UP F
2938 73B6            STW   FUNG   CLEAR PRINT FAULT UNKN F
2939 73B7            STW   FWAG   CLEAR RETYPE F
293A 739E            STW   RNTIME   CLEAR RUNNING TIME
293B 93B0            LDX   C9RG   CLEAR RETYPE WORDS
293C 7BB8            STW * HWA1
293D 0501            DXS   1
293E 113C            JMP   $-2
293F 0080            SMB   TIME
2940 8021            LDW   TIME   SYSTEM TIME
2941 A3B1            ADD   DERG   DELTA TIME
2942 73C5            STW   TIRG   CURRENT TIME
2943 0100  SM52      CLR
2944 739D            STW   BCRG   CLEAR 1 MIN COUNTER
2945 73B3            STW   FRRG   ZEVO RAMP FLAG
2946 214D            JSX   ZERG
2947 0B8F            D     FIRY   IN HVP
2948 214D            JSX   ZERG
```

```
          ' FAULT MONITOR  #9
2949 0B88            D      FTTY     IN HVP
294A 214D            JSX    ZERG
294B 0BAB            D      FNVY     IN HVP
294C 1133            JMP    SM53
294D 60EB      ZERG  STX    ERG-1
294E 9800            LDX *  0
294F 0100            CLR
2950 7800            STW *  0
2951 7801            STW *  1
2952 7802            STW *  2
2953 7803            STW *  3
2954 7804            STW *  4
2955 7805            STW *  5
2956 7806            STW *  6
2957 90EB            LDX    ERG-1
2958 1801            JMP *  1
2959 00A0      SM01  MSK
295A 200B            JSX    STOP
295B 0085      UF01  SMB    TYPEFLG   AND UF DATA LOG
295C 8617            LDW    TYPEFLG   INITIAL MESSAGE
295D 0800            SAZ              OUTPUT PATCH
295E 1160            JMP    $+2
295F 1166            JMP    UF02
2960 0100            CLR
2961 205B            JSX    PRRG
2962 5484            D      /M9RG
2963 0100            CLR
2964 0085            SMB    TYPEFLG
2965 7617            STW    TYPEFLG
```

```
               ' FAULT MONITOR  #10
               *
     2966      UF02      EQU  $
2966 0086                SMB  MODE    MODE MESSAGE PRINT
2967 3002                LDW  MODE
2968 F19C                CMW  FMODE    GET MODE -- IF NOT
2969 0870                SNE              SAME AS PREVIOUS MODE
296A 1176                JMP  FMTST       STORE AS LAST MODE
296B 0040                SLM              AND PRINT MESSAGE
296C 719C                STW  FMODE
296D 0130                CAX
296E 8996                LDW * MMSGS
296F 7175                STW  MMSG
2970 8003                LDW  ITYP PERMIT PRINT CHECK
2971 0800                SAZ
2972 1176                JMP  NMDP  SKIP PRINT
2973 0100                CLR
2974 205B                JSX  PRRG
2975 546E      MMSG      D    /M6RG
     2976      NMDP      EQU  $
2976 819C      FMTST     LDW  FMODE  IN MODES 2 AND 3 IF
2977 F3AB                CMW  CIRG   K51F HAS CHANGED
2978 0880                SGR         PRINT MESSAGE AND
2979 118C                JMP  UF03
297A 0086                SMB  K51F      STORE AS PREVIOUS
297B 800C                LDW  K51F      BAND MASK WORD
297C F19D                CMW  FBMSK
297D 0870                SNE
297E 118C                JMP  UF03
297F 719D                STW  FBMSK
2980 0130                CAX
2981 0401                IXS  1
2982 0A10                NOP
2983 0040                SLM
2984 899A                LDW * BMSGS
2985 718B                STW  BMSG
2986 8003                LDW  ITYP
2987 0800                SAZ
2988 118C                JMP  UF03
2989 0100                CLR
298A 205B                JSX  PRRG
298B 546E      BMSG      D    /M6RG
298C 8002      UF03      LDW  TMREQ
298D 0800                SAZ
298E 1190                JMP  $+2
298F 1194                JMP  UF035
2990 0100                CLR
2991 7002                STW  TMREQ
2992 205B                JSX  PRRG
2993 54A4                D    /MCRG
2994 83B2      UF035     LDW  FERG
2995 1129                JMP  SM50+1
2996 543E      MMSGS     D    /M2RG
2997 544A                D    /M3RG
2998 5456                D    /M4RG
2999 5460                D    /M5RG
299A 549A      BMSGS     D    /MBRG
```

```
299B 5490    484            D      /MARG
299C 0000    485 FMODE      DATA   0
299D 0000    486 FBMSK      DATA   0
             487
```



```
             488 ' FAULT MONITOR  #11
             489 *
             490 * BINARY TO DECIMAL CONVERSION
             491 * CALL WITH BINARY IN ACC
             492 *
299E 0000
299F 619E    493 CNVBD      SUBR
29A0 0810    494            SAP
29A1 11A4    495            JMP    MAXIT
29A2 F5D4    496            CMW    =9999    BCD RETURNED IN ACC
29A3 0890    497            SLE
     29A4    498 MAXIT      EQU    $
29A4 85D4    499            LDW    =9999
29A5 71C4    500            STW    CNVA
29A6 93A2    501            LDX    CCLG
29A7 61C5    502            STX    CNVC
29A8 93AC    503            LDX    C3RG
29A9 61C6    504            STX    CNVI
29AA 93A2    505 CNV1       LDX    CCLG
29AB 61C7    506            STX    CNVJ
29AC 91C6    507 CNV2       LDX    CNVI
29AD 0040    508            SLM
29AE F9C8    509            CMW *  CNVD
29AF 0340    510            SLS
29B0 11B2    511            JMP   $+2
29B1 11B8    512            JMP    CNV3
29B2 B9C8    513            SUB *  CNVD
29B3 91C7    514            LDX    CNVJ
29B4 0401    515            IXS    1
29B5 0000    516            HLT
29B6 61C7    517            STX    CNVJ
29B7 11AC    518            JMP    CNV2
29B8 71C4    519 CNV3       STW    CNVA
29B9 81C5    520            LDW    CNVC
29BA 0A14    521            SLL    4
29BB A1C7    522            ADD    CNVJ
29BC 0501    523            DXS    1
29BD 11C0    524            JMP    CNV4
29BE 919E    525            EXIT   CNVBD
29BF 2800
29C0 71C5    526 CNV4       STW    CNVC
29C1 81C4    527            LDW    CNVA
29C2 61C6    528            STX    CNVI
29C3 11AA    529            JMP    CNV1
29C4 0000    530 CNVA       D      0
29C5 0000    531 CNVC       D      0
29C6 0000    532 CNVI       D      0
29C7 0000    533 CNVJ       D      0
29C8 0001    534 CNVD       D      1
29C9 000A    535            D      10
29CA 0064    536            D      100
29CB 03E8    537            D      1000
             538
```

```
                 ' FAULT MONITOR  #12
                 *
                 * SUBROUTINE FOR COMPUTING SECOND COUNT
                 *
29CC 0000
29CD 61CC        SECNT    SUBR
29CE 0100                 CLR
29CF 71E6                 STW   MINADD
29D0 83C5                 LDW   TIRG
29D1 0080                 SMB   TIME
29D2 B021                 SUB   TIME
29D3 0110                 CMP
29D4 0820                 SAM
29D5 11E3                 JMP   PASTIME
29D6 A3B1                 ADD   DERG
29D7 93A2        REDO     LDX   CCLG
29D8 F5D5        LOOP     CMW   =500
29D9 0840                 SLS
29DA 11DC                 JMP   $+2
29DB 11E0                 JMP   DONE
29DC B5D5                 SUB   =500
29DD 0401                 IXS   1
29DE 0000                 HLT
29DF 11D8                 JMP   LOOP
29E0 0140        DONE     CXA
29E1 91CC                 EXIT  SECNT
29E2 2800
29E3 93AB        PASTIME  LDX   CIRG
29E4 61E6                 STX   MINADD
29E5 11D7                 JMP   REDO
29E6 0000        MINADD   D     0
```

```
                ' FAULT MONITOR  #14
2A0F B1B2       BING      D      X'B1B2'    1,2
2A10 B3B4                 D      X'B3B4'    3,4
2A11 B5B6                 D      X'B5B6'    5,6
2A12 B7B8                 D      X'B7B8'    7,8
2A13 B9C1                 D      X'B9C1'    9,A
2A14 B000                 D      X'B000'    0
     2A15       TMSTMP    EQU    $
2A15 B0B0       MINLC     TEXT   '000 '
2A16 B0A0
2A17 B0B0       SECLC     TEXT   '00 :'
2A18 A0BA
                *
2A19 C3CC       M1RG      TEXT   'CLOCK OUT/:'
2A1A CFC3
2A1B CBA0
2A1C CFD5
2A1D D4AF
2A1E BAA0
                *
2A1F C9CE       M2RG      TEXT   'INIT MODE/:'
2A20 C9D4
2A21 A0CD
2A22 CFC4
2A23 C5AF
2A24 BAA0
                *
2A25 D3D4       M3RG      TEXT   'STOP METER/:'
2A26 CFD0
2A27 A0CD
2A28 C5D4
2A29 C5D2
2A2A AFBA
                *
2A2B D3D4       M4RG      TEXT   'STOP GAP/:'
2A2C CFD0
2A2D A0C7
2A2E C1D0
2A2F AFBA
                *
2A30 CDCF       M5RG      TEXT   'MOVING MERGE/:'
2A31 D6C9
2A32 CEC7
2A33 A0CD
2A34 C5D2
2A35 C7C5
2A36 AFBA
                *
2A37 C6C1       M6RG      TEXT   'FAULT UNK/:'
2A38 D5CC
2A39 D4A0
2A3A D5CE
2A3B CBAF
2A3C BAA0
                *
2A3D D0D7       M7RG      TEXT   'PWR FAIL/:'
2A3E D2A0
```

```
              ' FAULT MONITOR  #13
              *
              *STORE CHARACTERS FOR BCD OR HEX VALUE
              *
29E7 0000
29E8 61E7     SDCHAR    SUBR
29E9 7203               STW    SDCA   PASSED IN ACC
29EA 8801               LDW *  1      CALLED WITH JSX
29EB 7205               STW    SDCE    NUMBER OF DIGITS -1 AT
29EC A800               ADD *  0       CALL+1  BYTE ADDR FOR 1ST
29ED 0130               CAX            DIGIT AT CALL+2
29EE 6206     SDBK      STX    SDCP    RETURNS TO CALL+3
29EF 8203               LDW    SDCA
29F0 E5D6               AND    =X'F'   GET DIGIT
29F1 0130               CAX
29F2 8203               LDW    SDCA    SET UP FOR NEXT DIGIT
29F3 0A04               SRL    4
29F4 7203               STW    SDCA
29F5 0040               SLM
29F6 5C0E               LDB *  /SDCC   GET CORRESPONDING CHAR.
29F7 9206               LDX    SDCP
29F8 0050               SGM
29F9 3800               STB *  0 STORE AT POINTER LOCATION
29FA 0501               DXS    1
29FB 11FD               JMP    $+2    UPDATE POINTER
29FC 0000               HLT
29FD 0140               CXA
29FE F205               CMW    SDCE
29FF 0840               SLS
2A00 11EE               JMP    SDBK
2A01 91E7               LDX    SDCHAR-1   FINISHED
2A02 2802               JSX *  2
2A03 0000     SDCA      D      0
2A04 0000     SDCB      D      0
2A05 0000     SDCE      D      0
2A06 0000     SDCP      D      0
2A07 B0B1     SDCC      D      X'B0B1'  0-9
2A08 B2B3               D      X'B2B3'
2A09 B4B5               D      X'B4B5'
2A0A B6B7               D      X'B6B7'
2A0B B8B9               D      X'B8B9'
2A0C C1C2               D      X'C1C2'  A-F
2A0D C3C4               D      X'C3C4'
2A0E C5C6               D      X'C5C6'
```

```
2A3F C6C1
2A40 C9CC
2A41 AFBA
              637 *
2A42 D4D9     638 M9RG      TEXT  'TYPE RUNID/:'
2A43 D0C5
2A44 A0D2
2A45 D5CE
2A46 C9C4
2A47 AFBA
              639 *
2A48 C2C1     640 MARG      TEXT  'BAND MSK/:'
2A49 CEC4
2A4A A0CD
2A4B D3CB
2A4C AFBA
              641 *
2A4D D5CE     642 MBRG      TEXT  'UNMASKED/:'
2A4E CDC1
2A4F D3CB
2A50 C5C4
2A51 AFBA
              643 *
2A52 D3C9     644 MCRG      TEXT  'SINCE INIT/:'
2A53 CEC3
2A54 C5A0
2A55 C9CE
2A56 C9D4
2A57 AFBA
              645 *
2A58 A0A0     646 MDRG      TEXT  '   REINIT/:'
2A59 A0D2
2A5A C5C9
2A5B CEC9
2A5C D4AF
2A5D BAA0
     2A58     647 MDBG      EQU   MDRG
              648 *
2A5E C6D5     649 MERG      TEXT  'FULL INIT/:'
2A5F CCCC
2A60 A0C9
2A61 CEC9
2A62 D4AF
2A63 BAA0
              650 *
2A64 AFBA     651 CRLF      TEXT  '/:/:'
2A65 AFBA
              652 *
2A66 C3D2     653 CFMSG     TEXT  'CRITICAL SENSOR FAILURE/:'
2A67 C9D4
2A68 C9C3
2A69 C1CC
2A6A A0D3
2A6B C5CE
2A6C D3CF
2A6D D2A0
2A6E C6C1
```

```
2A6F C9CC
2A70 D5D2
2A71 C5AF
2A72 BAA0
            654 *
2A73 D3C5   655 NCFMSG   TEXT  'SENSOR FAILURE/:'
2A74 CED3
2A75 CFD2
2A76 A0C6
2A77 C1C9
2A78 CCD5
2A79 D2C5
2A7A AFBA
            656
```

```
                657 'FAULT MONITOR  #15
                658 *
2A7B A0A0       659 FWY       TEXT  '  F7A:  F7B:  F6A:  F6B:'
2A7C C6B7
2A7D C1BA
2A7E A0A0
2A7F C6B7
2A80 C2BA
2A81 A0A0
2A82 C6B6
2A83 C1BA
2A84 A0A0
2A85 C6B6
2A86 C2BA
2A87 A0A0       660           TEXT  '  F5A:  F5B:  F4A:  F4B:'
2A88 C6B5
2A89 C1BA
2A8A A0A0
2A8B C6B5
2A8C C2BA
2A8D A0A0
2A8E C6B4
2A8F C1BA
2A90 A0A0
2A91 C6B4
2A92 C2BA
2A93 A0A0       661           TEXT  '  F3A:  F3B:  F2A:  F2B:'
2A94 C6B3
2A95 C1BA
2A96 A0A0
2A97 C6B3
2A98 C2BA
2A99 A0A0
2A9A C6B2
2A9B C1BA
2A9C A0A0
2A9D C6B2
2A9E C2BA
2A9F A0A0       662           TEXT  '  F1A:  F1B:'
2AA0 C6B1
2AA1 C1BA
2AA2 A0A0
2AA3 C6B1
2AA4 C2BA
2AA5 A0A0       663 FWY1      TEXT  '  F0A:  F0B:'
2AA6 C6B0
2AA7 C1BA
2AA8 A0A0
2AA9 C6B0
2AAA C2BA
2AAB A0A0       664           TEXT  '  F4C:  F4D:  F1C:  F0C:'
2AAC C6B4
2AAD C3BA
2AAE A0A0
2AAF C6B4
2AB0 C4BA
2AB1 A0A0
```

```
2AB2 C6B1
2AB3 C3BA
2AB4 AJAJ
2AB5 C6BJ
2AB6 C3BA
2AB7 AJAJ   665            TEXT  '  FJD:  F4E:  F4F:  F1D:'
2AB8 C6BJ
2AB9 C4BA
2ABA AJAJ
2ABB C6B4
2ABC C5BA
2ABD AJAJ
2ABE C6B4
2ABF C6BA
2ACJ AJAJ
2AC1 C6B1
2AC2 C4BA
2AC3 AJAJ   666            TEXT  '  FJE:  FJF:'
2AC4 C6BJ
2AC5 C5BA
2AC6 AJAJ
2AC7 C6BJ
2AC8 C6BA
            667 *
2AC9 AJAJ   668 MRG        TEXT  '   M1:   M2:   M3:   M4:'
2ACA AJCD
2ACB B1BA
2ACC AJAJ
2ACD AJCD
2ACE B2BA
2ACF AJAJ
2ADJ AJCD
2AD1 33BA
2AD2 AJAJ
2AD3 AJCD
2AD4 B4BA
2AD5 AJAJ   669            TEXT  '   M5:   M6:   M7:   M8:'
2AD6 AJCD
24D7 B5BA
2AD8 AJAJ
2AD9 AJCD
2ADA B6BA
2ADB AJAJ
2ADC AJCD
2ADD B7BA
2ADE AJAJ
2ADF AJCD
2AEJ B8BA
            67J *
2AE1 AJAJ   671 RMP        TEXT  '  R11:  R1J:   R9:   R8:'
2AE2 D2B1
2AE3 B1BA
2AE4 AJAJ
2AE5 D2B1
2AE6 BJBA
2AE7 AJAJ
2AE8 AJD2
```

```
2AE9 B9BA
2AEA A0A0
2AEB A0D2
2AEC B8BA
2AED A0A0   672           TEXT  '   R7:  R6B:  R6A:   R5:'
2AEE A0D2
2AEF B7BA
2AF0 A0A0
2AF1 D2B6
2AF2 C2BA
2AF3 A0A0
2AF4 D2B6
2AF5 C1BA
2AF6 A0A0
2AF7 A0D2
2AF8 B5BA
2AF9 A0A0   673           TEXT  '   R4:   R3:   R2:   R1:'
2AFA A0D2
2AFB B4BA
2AFC A0A0
2AFD A0D2
2AFE B3BA
2AFF A0A0
2B00 A0D2
2B01 B2BA
2B02 A0A0
2B03 A0D2
2B04 B1BA
            674 *
            675
```

```
               676 'FAULT MONITOR  #16
               677 *
2B05           678 SW0       RES   4
2B09           679 SWD       RES   4
2B0D           680 SWF       RES   4
               681 *
2B11           682 CSF0      RES   4
2B15           683 CSFD      RES   4
2B19           684 CSF       RES   4
               685 *
2B1D           686 SWI       RES   4
2B21           687 SWA       RES   4
2B25           688 SWH       RES   4
               689 *
2B29           690 FHC       RES   26   46 HOUR COUNTS
2B43           691 MHC       RES   8
2B4B           692 RHC       RES   12
               693 *
2B57           694 FC        RES   26   46 SUMMARY COUNT
2B71           695 MC        RES   8
2B79           696 RC        RES   12
               697 *
2B85           698 SAI       RES   1
2B86           699 BCNT      RES   1
2B87           700 PSI       RES   1
2B88 A0B0      701 NUMR      TEXT  ' 0000:'
2B89 B0B0
2B8A B0BA
2B8B           702 SCR2      RES   1
2B8C 8800      703 LDWX      LDW * 0
2B8D           704 STPRNT    RES   0
               705           DO    1,8
2B8D A0A0      706           D     '  '
2B8E A0A0
2B8F A0A0
2B90 A0A0
2B91 A0A0
2B92 A0A0
2B93 A0A0
2B94 A0A0
2B95 AFBA      707           D     '/:'
2B96           708 SCR3      RES   1
2B97           709 SCR4      RES   1
2B98           710 SCR5      RES   1
2B99           711 FAB       RES   1
2B9A           712 MAB       RES   1
2B9B           713 RAB       RES   1
               714
```

```
              ' FAULT MONITOR  #17
              * DATA BASE
2B9C 000A     ACRG      D      X'A'    LOOP TOLERANCE
2B9D 0002     BCRG      D      2    1 MIN COUNTER
2B9E 0000     RNTIME    D      0    RUNNING 1 MINUTE COUNTER
2B9F 0000     BC5G      D      0    10 MIN. COUNTER
2BA0 000A     CARG      D      10
2BA1 A000     CBLG      D      X'A000'   BLANK
2BA2 0000     CCLG      D      0    ZERO
2BA3 C200     CDRG      D      X'C200'   B
2BA4 FFFF     CFFG      D      X'FFFF'
2BA5 FFFE     CFEG      D      X'FFFE'
2BA6 8022     CWRD11    D      X'8022'   SIGN AND SIGNAL WORD
2BA7 0301     CDOT      D      X'0301'   DOT 0,1 INSTRUCTION
2BA8 000B     CIXG      D      11   NO. PRINT CHAR
2BA9 FFFE     CM2G      D      X'FFFE'   -2
2BAA C100     CURG      D      X'C100'   A
2BAB 0001     C1RG      D      1
2BAC 0003     C3RG      D      3
2BAD 0004     C4RG      D      4
2BAE 0006     C6RG      D      6
2BAF 0008     C8RG      D      8
2BB0 0009     C9RG      D      9
2BB1 7530     DERG      D      X'7530'   DELTA TIME 1 MIN
2BB2 0000     FERG      D      0    PREVIOUS FAULT FLAG
2BB3 0000     FRRG      D      0    ZERO RAMP TABLE FLAG
2BB4 0000     FSTG      D      0    START UP FLAG
2BB5 0000     FTRG      D      0    PRINT FLAG
2BB6 0000     FUNG      D      0    PRINT FAULT UNKN FLAG
2BB7 0000     FWAG      D      0    RETYPE FLAG
2BB8          HWA1      RES    4    HIGHW RETYPE
2BBC          HWA2      RES    6    RAMP RETYPE
2BC2 0027     LKRG      D      RERG   READ ADDRESS OP MON
2BC3 284A     PNRG      D      PTRG   PRINT ADDRESS
2BC4 003C     RESG      D      60   RESET TIME
2BC5 0000     TIRG      D      0    NEXT MINUTE TIME
2BC6 0014     WTRG      D      X'14'   SENSOR PAIR TOLERANCE
```

```
                 'FAULT MONITOR  #18
                 *
                 * PRINT SENSOR TABLE
                 *
                 *  FORMAT MATCHES BIT PATTERN OF INPUT COMMANDS
                 *     SEE OPERATOR MANUAL
                 *
                 *   CALL:
                 *          PST
                 *(0)       D      ADDRESS 4 WORDS OF SENSOR INFO
                 *(1)       RETURN
                 *
2BC7 0000
2BC8 63C7        PST        SUBR
2BC9 8800                   LDW * 0
2BCA A5D7                   ADD    =4
2BCB 95D8                   LDX    =-4
2BCC E5D9                   AND    =2047
2BCD C38C                   ORI    LDWX
2BCE 73D1                   STW    LDW2
                 *
     2BCF        PST10      EQU    $
2BCF 6396                   STX    SCR3
2BD0 0040                   SLM
2BD1 0A10        LDW2       NOP
2BD2 95DA                   LDX    =-16
                 *
     2BD3        PST20      EQU    $
2BD3 7397                   STW    SCR4
2BD4 06B0                   LLB    '0'
2BD5 0810                   SAP
2BD6 06C6                   LLB    'F'
2BD7 0040                   SLM
2BD8 3F2A                   STB *  STPRNT+16
2BD9 8397                   LDW    SCR4
2BDA 0A11                   SLL    1
2BDB 0401                   IXS    1
2BDC 13D3                   JMP    PST20
                 *
2BDD 0100                   CLR
2BDE 206F                   JSX    TYPO
2BDF 571A                   D      /STPRNT
                 *
2BE0 9396                   LDX    SCR3
2BE1 0401                   IXS    1
2BE2 13CF                   JMP    PST10
                 *
2BE3 93C7                   EXIT    PST,1
2BE4 2801
```

```
                'FAULT MONITOR  #19
                *
                * PRINT COUNTS
                *
                *    CALL:
                *         JSX   PRCNT  (LOCAL PAGE)
                *(0)      RETURN
                *
                *  PRINTS FREEWAY, RAMP, MERGE COUNTS
                *   FOR HOUR AND SUMMARY.
                *
2BE5 0000
2BE6 63E5       PRCNT     SUBR
2BE7 85DB                 LDW   =1
2BE8 206F                 JSX   TYPO
2BE9 54C8                 D     /CRLF
2BEA 0100                 CLR
2BEB 205B                 JSX   PRRG
2BEC 54C8                 D     /CRLF
                *
2BED 85DC                 LDW   =13  FREEWAY COUNTS
2BEE 206F                 JSX   TYPO
2BEF 54F6                 D     /FWY
2BF0 0100                 CLR
2BF1 206F                 JSX   TYPO
2BF2 54C8                 D     /CRLF
2BF3 85DD                 LDW   =14
2BF4 2438                 JSX   PCNTR
2BF5 2B37                 D     FHC+14
2BF6 0100                 CLR
2BF7 206F                 JSX   TYPO
2BF8 54C8                 D     /CRLF
2BF9 85DD                 LDW   =14
2BFA 2438                 JSX   PCNTR
2BFB 2B65                 D     FC+14
                *
2BFC 85DB                 LDW   =1
2BFD 206F                 JSX   TYPO
2BFE 54C8                 D     /CRLF
2BFF 85DE                 LDW   =11
2C00 206F                 JSX   TYPO
2C01 554A                 D     /FWY1
2C02 0100                 CLR
2C03 206F                 JSX   TYPO
2C04 54C8                 D     /CRLF
2C05 85DF                 LDW   =12
2C06 2438                 JSX   PCNTR
2C07 2B43                 D     FHC+26
2C08 0100                 CLR
2C09 206F                 JSX   TYPO
2C0A 54C8                 D     /CRLF
2C0B 85DF                 LDW   =12
2C0C 2438                 JSX   PCNTR
2C0D 2B71                 D     FC+26
                *
2C0E 85DB                 LDW   =1   RAMP COUNTS
2C0F 206F                 JSX   TYPO
```

```
2C10 54C8          D      /CRLF
2C11 85DE          LDW    =11
2C12 206F          JSX    TYPO
2C13 55C2          D      /RMP
2C14 0100          CLR
2C15 206F          JSX    TYPO
2C16 54C8          D      /CRLF
```



```
               'FAULT MONITOR  #20
               *
2C17 85DF          LDW    =12
2C18 2438          JSX    PCNTR
2C19 2B57          D      RHC+12
2C1A 0100          CLR
2C1B 206F          JSX    TYPO
2C1C 54C8          D      /CRLF
2C1D 85DF          LDW    =12
2C1E 2438          JSX    PCNTR
2C1F 2B85          D      RC+12
               *
2C20 85DB          LDW    =1         MERGE COUNTS
2C21 206F          JSX    TYPO
2C22 54C8          D      /CRLF
2C23 85E0          LDW    =7
2C24 206F          JSX    TYPO
2C25 5592          D      /MRG
2C26 0100          CLR
2C27 206F          JSX    TYPO
2C28 54C8          D      /CRLF
2C29 85E1          LDW    =8
2C2A 2438          JSX    PCNTR
2C2B 2B4B          D      MHC+8
2C2C 0100          CLR
2C2D 206F          JSX    TYPO
2C2E 54C8          D      /CRLF
2C2F 85E1          LDW    =8
2C30 2438          JSX    PCNTR
2C31 2B79          D      MC+8
               *
2C32 85DB          LDW    =1
2C33 206F          JSX    TYPO
2C34 54C8          D      /CRLF
2C35 93E5          EXIT   PRCNT
2C36 2800
```

```
               'FAULT MONITOR  #21
               *
               *  CONVERT AND PRINT LINE OF COUNTS
               *
               *   CALL:
               *          JSX    PCNTR  (LOCAL PAGE)
               *(0)       D      ADDRESS END OF COUNTS+1
               *(1)       RETURN
               *
               *  ACCUMULATOR--NUMBER OF WORDS PRECEDING ADDRESS
               *
2C37 0000
2C38 6437      PCNTR      SUBR
2C39 0110                 CMP
2C3A 7385                 STW    SA1
2C3B 8800                 LDW *  0
2C3C E5D9                 AND    =2047
2C3D C38C                 ORI    LDWX
2C3E 7442                 STW    LDW1
2C3F 9385                 LDX    SA1
     2C40      CONT       EQU    $
2C40 6385                 STX    SA1
2C41 0040                 SLM
2C42 0A10      LDW1       NOP
2C43 219F                 JSX    CNVBD
2C44 21E8                 JSX    SDCHAR
2C45 0003                 D      3
2C46 5711                 D      /NUMR+1
2C47 0100                 CLR
2C48 206F                 JSX    TYPO
2C49 5710                 D      /NUMR
2C4A 9385                 LDX    SA1
2C4B 0401                 IXS    1
2C4C 1440                 JMP    CONT
2C4D 9437                 EXIT   PCNTR,1
2C4E 2801
```

```
                936 'FAULT MONITOR  #22
                937 *
                938 * FAULTS
                939 *
                940 * CRITICAL FAULTS
                941 *
  2C4F 0000
  2C50 644F     942 CFAULT   SUBR
  2C51 95E2     943          LDX    =2
  2C52 0040     944          SLM
       2C53     945 CFL      EQU    $
  2C53 831D     946          LDW  * SWI
  2C54 EB11     947          AND  * CSFO   CRITICAL STUCK ON
  2C55 7B11     948          STW  * CSFO
  2C56 8B1D     949          LDW  * SWI
  2C57 0120     950          INV
  2C58 EB15     951          AND  * CSFD   CRITICAL DEAD
  2C59 7B15     952          STW  * CSFD
  2C5A CB11     953          ORI  * CSFO
  2C5B 7B19     954          STW  * CSF
  2C5C 0501     955          DXS    1
  2C5D 1453     956          JMP    CFL
                957 *
                958 * CHECK TIME
                959 *
  2C5E 003F     960          CUSC
S 2C5F 26E8
  2C60 0000     961 CCLK     D      0
  2C61 0000     962 CCNT     D      0
  2C62 003C     963 CTIM     D      60   SEC INTERVAL
  2C63 1465     964          JMP    $+2
  2C64 14B8     965          JMP    CFCXT
                966 *
                967 * GROUP SUMMARIES
                968 *
  2C65 0100     969          CLR
  2C66 7399     970          STW    FAB
  2C67 739B     971          STW    RAB
  2C68 739A     972          STW    MAB
  2C69 8319     973          LDW    CSF  FREEWAY SENSORS
  2C6A E5E3     974          AND    =16383
  2C6B F5E3     975          CMW    =16383
  2C6C 0360     976          SEQ
  2C6D 7399     977          STW    FAB   FREEWAY ALL INDICATE BAD
                978 *
  2C6E 831B     979          LDW    CSF+2  MERGE AREA
  2C6F E5E4     980          AND    =32640
  2C70 F5E4     981          CMW    =32640
  2C71 0360     982          SEQ
  2C72 739A     983          STW    MAB
                984 *
  2C73 831B     985          LDW    CSF+2   RAMP
  2C74 E5D7     986          AND    =4
  2C75 7398     987          STW    SCR5
  2C76 831A     988          LDW    CSF+1
  2C77 E5E5     989          AND    =4083
  2C78 C398     990          ORI    SCR5
```

```
2C79 7398          STW   SCR5
2C7A 8319          LDW   CSF
2C7B E5E6          AND   =X'8000'
2C7C C398          ORI   SCR5
2C7D F5E7          CMW   =-28681
2C7E 0860          SEQ
2C7F 739B          STW   RAB
```



```
               'FAULT MONITOR #23
               *
               * RAMP CRITICAL COMBINATIONS
               *
2C80 839B          LDW   RAB   IF ALL BAD ASSUME
2C81 0800          SAZ           DUE TO LACK OF TRAFFIC
2C82 1484          JMP   $+2
2C83 14A2          JMP   MCFAULT
               *
               * 1 R3&R4 YIELD SIGN CONTROL
               *
2C84 831A          LDW   CSF+1
2C85 E5E8          AND   =128
2C86 7398          STW   SCR5
2C87 831B          LDW   CSF+2
2C88 E5D7          AND   =4
2C89 C398          ORI   SCR5
2C8A F5E9          CMW   =132
2C8B 0870          SNE
2C8C 14BA          JMP   CFAIL
               *
               * 3 R6B+R6A STOP LIGHT CONTROL
               *
     2C8D      CF3CHK    EQU   $
2C8D 831A          LDW   CSF+1
2C8E E5EA          AND   =1536
2C8F 0800          SAZ
2C90 14BA          JMP   CFAIL
```

```
                 'FAULT MONITOR  #24
                 *
                 * 5 R11 NO RAMP TRAFFIC MASK
                 *
          2C91   CF5CHK    EQU   $
2C91 831A                  LDW   CSF+1
2C92 0830                  SAO
2C93 1495                  JMP   CF6CHK  NEXT CHECK
2C94 14BA                  JMP   CFAIL
                 *
                 * 6 R1, R2, R3 (2 OF 3) COUNT
                 *
          2C95   CF6CHK    EQU   $
2C95 831A                  LDW   CSF+1
2C96 E5EB                  AND   =2304
2C97 7398                  STW   SCR5
2C98 831B                  LDW   CSF+2
2C99 E5D7                  AND   =4
2C9A C398                  ORI   SCR5
2C9B F5EB                  CMW   =2304
2C9C 0860                  SEQ
2C9D F5EC                  CMW   =2052
2C9E 0860                  SEQ
2C9F F5ED                  CMW   =260
2CA0 0870                  SNE
2CA1 14BA                  JMP   CFAIL
                 *
                 * 4 M1-M5 MERGE AREA CONTROL
                 *
          2CA2   MCFAULT   EQU   $
2CA2 839A                  LDW   MAB
2CA3 0800                  SAZ
2CA4 14A6                  JMP   $+2
2CA5 14AA                  JMP   FCFAULT
2CA6 831B                  LDW   CSF+2
2CA7 E5EE                  AND   =3968
2CA8 0800                  SAZ
2CA9 14BA                  JMP   CFAIL
                 *
                 * 2 F4A-F1B FREEWAY SENSOR
                 *
          2CAA   FCFAULT   EQU   $
2CAA 8399                  LDW   FAB
2CAB 0800                  SAZ
2CAC 14AE                  JMP   $+2
2CAD 14B2                  JMP   CFCND
2CAE 8319                  LDW   CSF
2CAF E5EF                  AND   =16320
2CB0 0800                  SAZ
2CB1 14BA                  JMP   CFAIL
                 *
          2CB2   CFCND     EQU   $
2CB2 85F0                  LDW   =-1
2CB3 95DE                  LDX   =11
2CB4 0040                  SLM
          2CB5   CSFCLR    EQU   $
2CB5 7B11                  STW * CSF0
```

```
     FAULT MONITOR  #24                                           PAGE  35

2CB6 0501                DXS    1
2CB7 14B5                JMP    CSFCLR
     2CB8      CFCXT     EQU    $
2CB8 944F                EXIT   CFAULT
2CB9 2800



     FAULT MONITOR  #25                                           PAGE  36

               'FAULT MONITOR  #25
               *
               * CRITICAL SENSOR HAS FAILED
               *
     2CBA      CFAIL     EQU    $
2CBA 8319                LDW    CSF
2CBB E5EF                AND    =16320
2CBC 7319                STW    CSF
               *
2CBD 831A                LDW    CSF+1
2CBE E5F1                AND    =3969
2CBF 731A                STW    CSF+1
               *
2CC0 831B                LDW    CSF+2
2CC1 E5F2                AND    =3972
2CC2 731B                STW    CSF+2
               *
2CC3 0100                CLR
2CC4 731C                STW    CSF+3
               *
2CC5 0100                CLR
2CC6 205B                JSX    PRRG
2CC7 54CC                D      /CFMSG
2CC8 23C8                JSX    PST
2CC9 2B19                D      CSF
2CCA 20DB                JSX    CDISP
2CCB 100B                JMP    STOP
```

```
                1118 'FAULT MONITOR  #26
                1119 *
                1120 * NON CRITICAL FAULTS
                1121 *
  2CCC 0000
  2CCD 64CC     1122 NCFAULT   SUBR
  2CCE 008F     1123           CUSC
S 2CCF 26E8
  2CD0 0000     1124 NCLK      D      0
  2CD1 0000     1125 NCNT      D      0
  2CD2 0384     1126 NTIM      D      900  15 MIN INTERVAL
  2CD3 14D5     1127           JMP    $+2
  2CD4 14F8     1128           JMP    NCFXT
                1129 *
  2CD5 3003     1130           LDW    ITYP
  2CD6 0800     1131           SAZ
  2CD7 14F2     1132           JMP    NCFND
                1133 *
  2CD8 830D     1134           LDW    SWF  SELECT BITS OF INTEREST
  2CD9 E5F3     1135           AND    =-16385
  2CDA 730D     1136           STW    SWF
  2CDB 738B     1137           STW    SCR2
                1138 *
  2CDC 830E     1139           LDW    SWF+1
  2CDD E5E5     1140           AND    =4083
  2CDE 730E     1141           STW    SWF+1
  2CDF C38B     1142           ORI    SCR2
  2CE0 738B     1143           STW    SCR2
                1144 *
  2CE1 830F     1145           LDW    SWF+2
  2CE2 E5F4     1146           AND    =32644
  2CE3 730F     1147           STW    SWF+2
  2CE4 C38B     1148           ORI    SCR2
  2CE5 738B     1149           STW    SCR2
                1150 *
  2CE6 8310     1151           LDW    SWF+3
  2CE7 E5F5     1152           AND    =-14338
  2CE8 7310     1153           STW    SWF+3
  2CE9 C38B     1154           ORI    SCR2
  2CEA 0800     1155           SAZ
  2CEB 14ED     1156           JMP    $+2
  2CEC 14F2     1157           JMP    NCFND
                1158 *
  2CED 0100     1159           CLR
  2CEE 205B     1160           JSX    PRRG
  2CEF 54E6     1161           D      /NCFMSG
  2CF0 23C8     1162           JSX    PST
  2CF1 2B0D     1163           D      SWF
       2CF2     1164 NCFND     EQU    $
  2CF2 85F0     1165           LDW    =-1
  2CF3 95DE     1166           LDX    =11
  2CF4 0040     1167           SLM
       2CF5     1168 NCFCL1    EQU    $
  2CF5 7B05     1169           STW  * SW0
  2CF6 0501     1170           DXS    1
  2CF7 14F5     1171           JMP    NCFCL1
       2CF8     1172 NCFXT     EQU    $
```

```
2CF8 94CC          EXIT   NCFAULT
2CF9 2800
```

FAULT MONITOR #27 PAGE 39

```
          'FAULT MONITOR  #27
          *
          * SENSOR INPUT
          *
2CFA 0000
2CFB 64FA  SINP      SUBR
2CFC 02F6            DIN    15,6   SENSOR WORDS 1
2CFD 25BF            JSX    STP
2CFE 0000            D      0
2CFF 02F5            DIN    15,5
2D00 25BF            JSX    STP
2D01 0001            D      1
2D02 02F3            DIN    15,3   3
2D03 25BF            JSX    STP
2D04 0002            D      2
2D05 02F7            DIN    15,7   4
2D06 25BF            JSX    STP
2D07 0003            D      3
2D08 008F            CUSC
S 2D09 26E8
2D0A 0000  ACLK      D      0
2D0B 0000  ACNT      D      0
2D0C 0E10  ATIM      D      3600  1 HR
2D0D 150F            JMP    $+2
2D0E 1519            JMP    NHR
2D0F 8003            LDW    ITYP IS TYPE-OUT ON?
2D10 0800            SAZ     SKIP IF GO
2D11 1513            JMP    $+2
2D12 23E6            JSX    PRCNT
2D13 95F6            LDX    =45
2D14 0100            CLR
2D15 0040            SLM
     2D16  ACLR      EQU    $
2D16 7B29            STW *  FHC  ZERO HOUR COUNTS
2D17 0501            DXS    1
2D18 1516            JMP    ACLR
     2D19  NHR       EQU    $
```

```
          ' FAULT MONITOR  #28
          *
          * ACTIVATIONS
          *    SENSOR COUNTS ORDERED SAME AS
          *    MSG SECTION
          *
2D19 95F7             LDX    =-14  FOR FREEWAY
2D1A 8321             LDW    SWA
2D1B 0040             SLM
          *
     2D1C   ACT14     EQU    $    F7A-F1B
2D1C 0830             SAO
2D1D 1526             JMP    NOBMP1
2D1E 7385             STW    SA1
2D1F 8B37             LDW  * FHC+14
2D20 A5DB             ADD    =1
2D21 7B37             STW  * FHC+14
2D22 8B65             LDW  * FC+14
2D23 A5DB             ADD    =1
2D24 7B65             STW  * FC+14
2D25 8385             LDW    SA1
     2D26   NOBMP1    EQU    $
2D26 0A01             SRL    1
2D27 0401             IXS    1
2D28 151C             JMP    ACT14
```

```
                 'FAULT MONITOR  #29
                 *
                 *
2D29 8324                  LDW   SWA+3   F0F
2D2A 258F                  JSX   FIT
2D2B 0019                  D     25
2D2C 0A11                  SLL   1       F0C
2D2D 258F                  JSX   FIT
2D2E 0013                  D     19
2D2F 0A14                  SLL   4       F0A
2D30 258F                  JSX   FIT
2D31 000E                  D     14
2D32 0A11                  SLL   1       F0B
2D33 258F                  JSX   FIT
2D34 000F                  D     15
2D35 0A11                  SLL   1       F4D
2D36 258F                  JSX   FIT
2D37 0011                  D     17
2D38 0A11                  SLL   1       F4C
2D39 258F                  JSX   FIT
2D3A 0010                  D     16
2D3B 0A11                  SLL   1       F4F
2D3C 258F                  JSX   FIT
2D3D 0016                  D     22
2D3E 0A11                  SLL   1       F4E
2D3F 258F                  JSX   FIT
2D40 0015                  D     21
2D41 0A11                  SLL   1       F1D
2D42 258F                  JSX   FIT
2D43 0017                  D     23
2D44 0A11                  SLL   1       F1C
2D45 258F                  JSX   FIT
2D46 0012                  D     18
2D47 0A11                  SLL   1       F0E
2D48 258F                  JSX   FIT
2D49 0018                  D     24
2D4A 0A11                  SLL   1       F0D
2D4B 258F                  JSX   FIT
2D4C 0014                  D     20
```

```
                'FAULT MONITOR  #30
                *
                *  FOR RAMP
                *
2D4D 8323                LDW   SWA+2  R3
2D4E 0A02                SRL   2
2D4F 259F                JSX   RIT
2D50 0009                D     9
2D51 8321                LDW   SWA    R9
2D52 0A51                SLC   1
2D53 259F                JSX   RIT
2D54 0002                D     2
2D55 8322                LDW   SWA+1  R11
2D56 259F                JSX   RIT
2D57 0000                D     0
2D58 0A01                SRL   1      R10
2D59 259F                JSX   RIT
2D5A 0001                D     1
2D5B 0A03                SRL   3      R8
2D5C 259F                JSX   RIT
2D5D 0003                D     3
2D5E 0A01                SRL   1      R7
2D5F 259F                JSX   RIT
2D60 0004                D     4
2D61 0A01                SRL   1      R5
2D62 259F                JSX   RIT
2D63 0007                D     7
2D64 0A01                SRL   1      R4
2D65 259F                JSX   RIT
2D66 0008                D     8
2D67 0A01                SRL   1      R2
2D68 259F                JSX   RIT
2D69 000A                D     10
2D6A 0A01                SRL   1      R6A
2D6B 259F                JSX   RIT
2D6C 0006                D     6
2D6D 0A01                SRL   1      R6B
2D6E 259F                JSX   RIT
2D6F 0005                D     5
2D70 0A01                SRL   1      R1
2D71 259F                JSX   RIT
2D72 000B                D     11
```

```
               'FAULT MONITOR  #31
               *
               * FOR MERGE AREA
               *
2D73 8323               LDW    SWA+2  M8
2D74 0A11               SLL    1
2D75 25AF               JSX    MIT
2D76 0007               D      7
2D77 0A11               SLL    1      M7
2D78 25AF               JSX    MIT
2D79 0006               D      6
2D7A 0A11               SLL    1      M6
2D7B 25AF               JSX    MIT
2D7C 0005               D      5
2D7D 0A11               SLL    1      M1
2D7E 25AF               JSX    MIT
2D7F 0000               D      0
2D80 0A11               SLL    1      M5
2D81 25AF               JSX    MIT
2D82 0004               D      4
2D83 0A11               SLL    1      M4
2D84 25AF               JSX    MIT
2D85 0003               D      3
2D86 0A11               SLL    1      M3
2D87 25AF               JSX    MIT
2D88 0002               D      2
2D89 0A11               SLL    1      M2
2D8A 25AF               JSX    MIT
2D8B 0001               D      1
2D8C 94FA               EXIT   SINP
2D8D 2800
```

```
                 *FAULT MONITOR  #32
                 *
                 * FREEWAY COUNT INCREMENTATION
                 *
                 *    CALL:
                 *         JSX   FIT  (LOCAL PAGE)
                 *(0)      D     0 RELATIVE INDEX TO COUNT WORD
                 *(1)      RETURN
                 *
                 *   ACCUMULATOR--SIGN BIT = 1 INDICATES ACTIVATION
                 *
2D8E 0000
2D8F 658E        FIT       SUBR
2D90 0820                  SAM     ACTIVATED?
2D91 2801                  JSX * 1  NO, BACK QUICK
2D92 7385                  STW   SA1
2D93 9800                  LDX * 0
2D94 0040                  SLM
2D95 8B29                  LDW * FHC
2D96 A5DB                  ADD   =1
2D97 7B29                  STW * FHC
2D98 8B57                  LDW * FC
2D99 A5DB                  ADD   =1
2D9A 7B57                  STW * FC
2D9B 8335                  LDW   SA1
2D9C 958E                  EXIT  FIT,1
2D9D 2801
                 *
                 * RAMP COUNT INCREMENTATION
                 *
                 *    CALL:
                 *         JSX   RIT (LOCAL PAGE)
                 *(0)      D     0 RELATIVE INDEX TO COUNT WORD
                 *(1)      RETURN
                 *
                 *  ACCUMULATOR--RIGHT END = 1 INDICATES ACTIVATION
                 *
2D9E 0000
2D9F 659E        RIT       SUBR
2DA0 0830                  SA0      ACTIVATED?
2DA1 2801                  JSX * 1  NOPE, BACK QUICK
2DA2 7335                  STW   SA1
2DA3 9800                  LDX * 0
2DA4 0040                  SLM
2DA5 8B4B                  LDW * RHC
2DA6 A5DB                  ADD   =1
2DA7 7B4B                  STW * RHC
2DA8 8B79                  LDW * RC
2DA9 A5DB                  ADD   =1
2DAA 7B79                  STW * RC
2DAB 8385                  LDW   SA1
2DAC 959E                  EXIT  RIT,1
2DAD 2801
```

```
          'FAULT MONITOR  #33
          *
          * MERGE COUNT INCREMENTATION
          *
          *          CALL:
          *          JSX   MIT  (LOCAL PAGE)
          *(0)       D     0 RELATIVE INDEX TO COUNT WORD
          *(1)       RETURN
          *
          *  ACCUMULATOR--SIGN BIT = 1 INDICATES ACTIVATION
          *
2DAE 0000
2DAF 65AE MIT        SUBR
2DB0 0820            SAM    ACTIVATED?
2DB1 2801            JSX * 1   NOPE, BACK QUICK
2DB2 7335            STW   SA1
2DB3 9800            LDX * 0
2DB4 0040            SLM
2DB5 8B43            LDW * MHC
2DB6 A5DB            ADD   =1
2DB7 7B43            STW * MHC
2DB8 8B71            LDW * MC
2DB9 A5DB            ADD   =1
2DBA 7B71            STW * MC
2DBB 8385            LDW   SA1
2DBC 95AE            EXIT  MIT,1
2DBD 2801
```

```
               'FAULT MONITOR  #34
               *
               * SENSOR INPUT PROCESSING
               *
               *   CALL:
               *          STP
               *(0)       D      0-RELATIVE INDEX TO SENSOR WORD
               *(1)       RETURN
               *
2DBE 0000
2DBF 65BE      STP        SUBR
2DC0 9800                 LDX * 0   INDEX
2DC1 0040                 SLM
2DC2 7B1D                 STW * SWI  INPUT TABLE
2DC3 EB05                 AND * SWO  STUCK ON
2DC4 7B05                 STW * SWO
2DC5 8B25                 LDW * SWH  ACTIVATIONS
2DC6 0120                 INV
2DC7 EB1D                 AND * SWI
2DC8 7B21                 STW * SWA
2DC9 8B1D                 LDW * SWI  INPUT NOW BECOMES HISTORY
2DCA 7B25                 STW * SWH
2DCB 0120                 INV        DEAD ONES
2DCC EB09                 AND * SWD
2DCD 7B09                 STW * SWD
2DCE CB05                 ORI * SWO  SUMMARY OF FAULTS
2DCF 7B0D                 STW * SWF
2DD0 95BE                 EXIT  STP,1
2DD1 2801
               *
               *
               *
                          END

2DD2 0067
2DD3 0017
2DD4 270F
2DD5 01F4
2DD6 000F
2DD7 0004
2DD8 FFFC
2DD9 07FF
2DDA FFF0
2DDB 0001
2DDC 000D
2DDD 000E
2DDE 000B
2DDF 000C
2DE0 0007
2DE1 0008
2DE2 0002
2DE3 3FFF
2DE4 7F80
2DE5 0FF3
2DE6 8000
2DE7 8FF7
2DE8 0080
```

```
2DE9 0084
2DEA 0600
2DEB 0900
2DEC 0804
2DED 0104
2DEE 0F80
2DEF 3FC0
2DF0 FFFF
2DF1 0F81
2DF2 0F84
2DF3 BFFF
2DF4 7F84
2DF5 C7FE
2DF6 002D
2DF7 FFF2
```

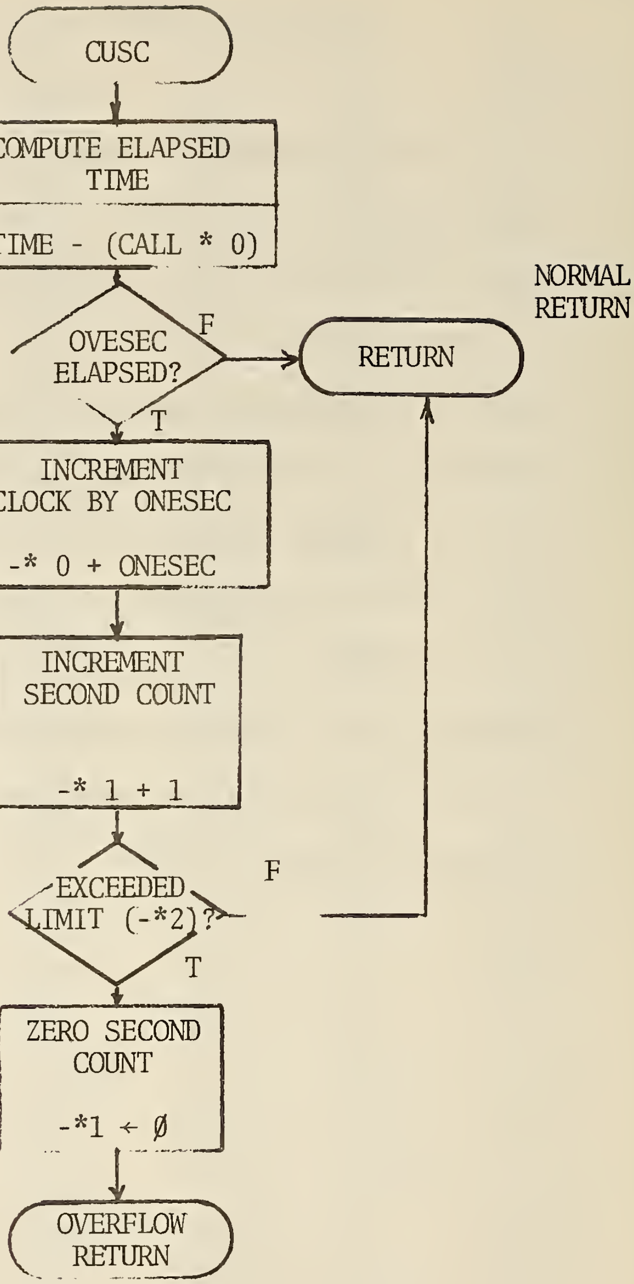
NO ERRORS

ACLK	2D0A	ACLR	2D16	ACNT	2D0B	ACRG	2B9C
ACT14	2D1C	ATIM	2D0C	BC5G	2B9F	BCNT	2B86
BCRG	2B9D	BEGT	0020	BING	2A0F	BMSG	298B
BMSGS	299A	C1RG	2BAB	C3RG	2BAC	C4RG	2BAD
C6RG	2BAE	C8RG	2BAF	C9RG	2BB0	CARG	2BA0
CBLG	2BA1	CCLG	2BA2	CCLK	2C60	CCNT	2C61
CDISP	28DB	CDOT	2BA7	CDRG	2BA3	CF3CHK	2C8D
CF5CHK	2C91	CF6CHK	2C95	CFAIL	2CBA	CFAULT	2C50
CFCND	2CB2	CFCXT	2CB8	CFEG	2BA5	CFFG	2BA4
CFL	2C53	CFMSG	2A66	CHLG	2805	CHSG	2806
CIXG	2BA8	CJ	2815	CM2G	2BA9	CNV1	29AA
CNV2	29AC	CNV3	29B8	CNV4	29C0	CNVA	29C4
CNVBD	299F	CNVC	29C5	CNVD	29C8	CNVI	29C6
CNVJ	29C7	CONT	2C40	CRLF	2A64	CRLG	2807
CRSG	2808	CSF	2B19	CSFCLR	2CB5	CSFD	2B15
CSFO	2B11	CTFG	2804	CTIM	2C62	CURG	2BAA
CUSC	3EE8	CWRD11	2BA6	DERG	2BB1	DONE	29E0
ERG	28EC	ERRCH	0046	FAB	2B99	FBMSK	299D
FC	2B57	FCFAULT	2CAA	FERG	2BB2	FHC	2B29
FIRY	0B8F	FIT	2D8F	FMODE	299C	FMTST	2976
FNVY	0BAB	FRRG	2BB3	FSTG	2BB4	FTRG	2BB5
FTTY	0B88	FUNG	2BB6	FWAG	2BB7	FWY	2A7B
FWY1	2AA5	HWA1	2BB8	HWA2	2BBC	IE01	28BA
IE06	28BD	IRRG	28A6	ITYP	2803	K51F	180C
LDW1	2C42	LDW2	2BD1	LDWX	2B8C	LKRG	2BC2
LOOP	29D8	M1RG	2A19	M2RG	2A1F	M3RG	2A25
M4RG	2A2B	M5RG	2A30	M6RG	2A37	M7RG	2A3D
M9RG	2A42	MAB	2B9A	MARG	2A48	MAXIT	29A4
MBRG	2A4D	MC	2B71	MCFAULT	2CA2	MCRG	2A52
MDBG	2A58	MDRG	2A58	MERG	2A5E	MHC	2B43
MINADD	29E6	MINLC	2A15	MIT	2DAF	MMSG	2975
MMSGS	2996	MODE	1802	MRG	2AC9	MSG	2895
NCFAULT	2CCD	NCFCL1	2CF5	NCFMSG	2A73	NCFND	2CF2
NCFXT	2CF8	NCLK	2CD0	NCNT	2CD1	NCTOUT	2920
NHR	2D19	NMDP	2976	NOBMP1	2D26	NOTND	2886
NTIM	2CD2	NUMR	2B88	NXTCH	2893	OUTCHAR	2899
PASTIME	29E3	PCDN	2857	PCHK	284F	PCNTR	2C38
PERG	2809	PFRTN	28C3	PLDM2	2839	PNC	287C
PNRG	2BC3	PR00	287C	PRCNT	2BE6	PREIN	2833
PRRG	285B	PS1	2B87	PST	2BC8	PST10	2BCF
PST20	2BD3	PTRG	284A	PURTN	28D1	RAB	2B9B
RC	2B79	REDO	29D7	RERG	0027	RESG	2BC4
RHC	2B4B	RIT	2D9F	RMP	2AE1	RNTIME	2B9E
SA1	2B85	SARG	2800	SCR2	2B8B	SCR3	2B96
SCR4	2B97	SCR5	2B98	SDBK	29EE	SDCA	2A03
SDCB	2A04	SDCC	2A07	SDCE	2A05	SDCHAR	29E8
SDCP	2A06	SECLC	2A17	SECNT	29CD	SERG	2801
SINP	2CFB	SJ	281F	SM00	2935	SM01	2959
SM02	2916	SM50	2928	SM51	292F	SM52	2943
SM53	2933	SORG	280A	SP00	2920	STAG	280F
STOP	280B	STP	2DBF	STPRNT	2B8D	SWA	2B21
SWD	2B09	SWF	2B0D	SWH	2B25	SWI	2B1D
SW0	2305	T1	2888	T2	2892	TIME	0021
TIRG	2BC5	TMREQ	2802	TMSTMP	2A15	TRAG	062C
TWAIT	289F	TYPEFLG	1617	TYPO	286F	UF01	295B
UF02	2966	UF03	298C	UF035	2994	WTRG	2BC6
ZERG	294D						
PAS?							

CHECK AND UPDATE SECOND COUNT SUBROUTINE

Provides a method of counting seconds to provide needed time keeping services beyond the 131.072 sec. limit imposed by present single word time without installing double word time calculations.

Upon entry the routine accesses the present system time and computes the elapsed time from the saved clock value in the call (TIME and SUB*Ø). The elapsed time is compared to counts of .002 sec. clock in a second (ONESEC) and if it is exceeded, the second count in the call is incremented (-*1) and compared against the limit (-*2). The clock value in the call is incremented by ONESEC and returned on second overflow and the second count is zeored out and the count overflow returns (JSX *3) taken for the count exceeded. The non-overflow return is JSX*4.


CUSC
COMPUTE ELAPSED TIME
TIME - (CALL * 0)
OVESEC ELAPSED?
F
T
RETURN
NORMAL RETURN
INCREMENT CLOCK BY ONESEC
-* 0 + ONESEC
INCREMENT SECOND COUNT
-* 1 + 1
EXCEEDED LIMIT (-*2)?
F
T
ZERO SECOND COUNT
-*1 ← Ø
OVERFLOW RETURN

```
CHECK AND UPDATE SECOND COUNT                          PAGE   1

           'CHECK AND UPDATE SECOND COUNT
           *
           *  CALL:
           *        CUSC
           *(0)     D      CLOCK
           *(1)     D      SECOND COUNT
           *(2)     D      LIMIT  RANGE 0-32767 SEC
           *(3)     RETURN FOR COUNT OVERFLOW
           *(4)     RETURN NORMAL
           *
           * CLOCK IS .002 SEC COUNT
           * WHEN 500 HAVE BEEN COUNTED SECOND COUNT
           *   INCREMENTED AND CHECKED AGAINST LIMIT
           * COUNT IS RESET AND OVERFLOW EXIT TAKEN IF SECOND
           *   COUNT EXCEEDS LIMIT
           * NOTE: ACCUMULATOR IS NOT SAVED
           *
```

```
CHECK AND UPDATE SECOND COUNT                          PAGE   2

           'CHECK AND UPDATE SECOND COUNT
     0021   TIME    EQU  X'21'
                    ORIG X'3EE8'
     3EE8   CUSC    EQU  $
3EE8 0080           SMB  TIME       SYSTEM TIME
3EE9 8021           LDW  TIME
3EEA B800           SUB * 0        -SAVED CLOCK AT LAST SEC
3EEB F6FC           CMW  ONESEC       >= 1 SEC?
3EEC 0840           SLS               NO
3EED 16EF           JMP  $+2       YES
3EEE 2804           JSX * 4
3EEF 8800           LDW * 0        BUMP CLOCK UP
3EF0 A6FC           ADD   ONESEC
3EF1 7800           STW * 0
3EF2 8801           LDW * 1         INCR SEC COUNT
3EF3 A6FD           ADD   ONE
3EF4 7801           STW * 1
3EF5 F802           CMW * 2         EXCEED LIMIT?
3EF6 0840           SLS               NO
3EF7 16F9           JMP   CUSCOV   YES
3EF8 2804           JSX * 4         NORMAL RETURN
     3EF9   CUSCOV  EQU   $
3EF9 0100           CLR              RESET SECOND COUNT
3EFA 7801           STW * 1
3EFB 2803           JSX * 3
3EFC 01F4   ONESEC    D  500
3EFD 0001   ONE       D  1
                    END


NO ERRORS
```

```
CUSC        3EE8  CUSCOV    3EF9  ONE         3EFD  ONESEC    3EFC
TIME        0021
PAS?
```

LOAD AND DUMP PROGRAM

To add the capability of bootstrapping a magnetic tape copy of the system the magnetic tape handling routine was modified so it would use only the right byte in each word for dumps and loads thereby creating a bootstrappable tape.

For magnetic tape load, the storage of data was changed to a byte instruction (STB*0) and the count modified accordingly (word count doubled).

For magnetic tape dump the load instruction was changed (LDB* 0) and the address check adjusted similar to those for the load.

The rest of the program remains the same and because the changes were so insignificant no flow chart is included for them.

```
                'LOAD AND DUMP PROGRAM
                *  VERSION PREPARED 7-18-74
                *  MODIFIED 12/28/76   JGB
                *     TO PRODUCE A BOOTSTRAPABLE MAG TAPE
                *     TO BOOT SET IX = X'0028'
                *        SS0 UP
                *        SS1 UP
                *        SS2 DOWN
                *        SS3 UP
                *   PRESS STEP THEN RUN
                *
                             ORIG   X'3F00'
     0020       EXEC         EQU    X'20'
3F00 8700       BEGIN        LDW    $
3F01 08C0                    SS0
3F02 176C                    JMP    DUMP
3F03 08D0       LOAD         SS1
3F04 1755                    JMP    MAGLOAD
3F05 1708                    JMP    PLOAD
3F06 0020       CORESTART D         X'20'
3F07 3FFF       COREND       D      X'3FFF'
3F08 03D0       PLOAD        DOT    X'D',0
3F09 0100                    CLR
3F0A 7751                    STW    COUNT
3F0B 03D9       LEADER       DOT    X'D',9
3F0C 02D0       STAT1        DIN    X'D',0
3F0D 0A17                    SLL    7
3F0E 0820                    SAM
3F0F 170C                    JMP    STAT1
3F10 02DD                    DIN    X'D',X'D'
3F11 0800                    SAZ
3F12 1714                    JMP    FRONT
3F13 170C                    JMP    STAT1
3F14 7753       FRONT        STW    CHAR
3F15 8751                    LDW    COUNT
3F16 A7EC                    ADD    =1
3F17 7751                    STW    COUNT
3F18 F7ED                    CMW    =4
3F19 0860                    SEQ
3F1A 1723                    JMP    FIVE
3F1B 8753       FOUR         LDW    CHAR
3F1C 36A4                    STB    /INDEX
3F1D 02D0       STAT2        DIN    X'D',0
3F1E 0A17                    SLL    7
3F1F 0820                    SAM
3F20 171D                    JMP    $-3
3F21 02DD                    DIN    X'D',X'D'
3F22 1714                    JMP    FRONT
```

```
                'LOAD AND DUMP PROGRAM
3F23 F7EE       FIVE      CMW    =5
3F24 0860                 SEQ
3F25 1729                 JMP    ELEVEN
3F26 8753                 LDW    CHAR
3F27 36A5                 STB    /INDEX+1
3F28 171D                 JMP    STAT2
3F29 F7EF       ELEVEN    CMW    =11
3F2A 0860                 SEQ
3F2B 172F                 JMP    TWELVE
3F2C 8753                 LDW    CHAR
3F2D 36A8                 STB    /LAST
3F2E 171D                 JMP    STAT2
3F2F F7F0       TWELVE    CMW    =12
3F30 8753                 LDW    CHAR
3F31 0870                 SNE
3F32 1736                 JMP    $+4
3F33 0840                 SLS
3F34 0000                 HLT
3F35 171D                 JMP    STAT2
3F36 36A9                 STB    /LAST+1
3F37 8752       SETADDR   LDW    INDEX
3F38 0A11                 SLL    1
3F39 0130                 CAX
3F3A A754                 ADD    LAST
3F3B B7F1                 SUB    =2
3F3C 7754                 STW    LAST
3F3D 0050                 SGM
3F3E 02D0       STAT3     DIN    X'D',0
3F3F 0A17                 SLL    7
3F40 0820                 SAM
3F41 173E                 JMP    $-3
3F42 02DD                 DIN    X'D',X'D'
3F43 3800                 STB *  0
3F44 0401                 IXS    1
3F45 0A10                 NOP
3F46 0140                 CXA
3F47 F754                 CMW    LAST
3F48 0860                 SEQ
3F49 173E                 JMP    STAT3
```

```
                 'LOAD AND DUMP PROGRAM
  3F4A 02D0      STAT4      DIN    X'D',0
  3F4B 0A17                 SLL    7
  3F4C 0820                 SAM
  3F4D 174A                 JMP    $-3
  3F4E 03D0                 DOT    X'D',0
  3F4F 0000                 HLT
  3F50 1700                 JMP    BEGIN
  3F51 0000      COUNT      D      0
  3F52 0000      INDEX      D      0
  3F53 0000      CHAR       D      0
  3F54 0000      LAST       D      0
       3F55      MAGLOAD    EQU    $
  3F55 0050                 SGM
  3F56 0390                 DOT    9,0
  3F57 976B                 LDX    FORTY
  3F58 0100                 CLR
  3F59 0399                 DOT    9,9
       3F5A      MAGRWT     EQU    $
  3F5A 0290                 DIN    9,0
  3F5B 0AC2                 SRC L  2
  3F5C 0810                 SAP
  3F5D 1766                 JMP    MAGRXT
  3F5E 0A11                 SLL    1
  3F5F 0820                 SAM
  3F60 175A                 JMP    MAGRWT
  3F61 029F                 DIN    9,15
  3F62 3800                 STB *  0
  3F63 0401                 IXS    1
  3F64 1766                 JMP    MAGRXT
  3F65 175A                 JMP    MAGRWT
       3F66      MAGRXT     EQU    $
  3F66 0100                 CLR
  3F67 0390                 DOT    9,0
  3F68 0002                 HLT    2
  3F69 0080                 EXEC
$ 3F6A 2020
  3F6B 0028      FORTY      D      40
```

```
                 * LOAD AND DUMP PROGRAM
3F6C 08D0        DUMP     SS1
3F6D 17B5                 JMP    MAGDMP
3F6E 97F2        PLEAD    LDX    =200
3F6F 03C0                 DOT    X'C',0
3F70 03C2                 DOT    X'C',2
3F71 0100        STAT8    CLR
3F72 03C6                 DOT    X'C',6
3F73 02C0                 DIN    X'C',0
3F74 0A17                 SLL    7
3F75 0820                 SAM
3F76 1773                 JMP    $-3
3F77 0501                 DXS    1
3F78 1771                 JMP    STAT8
3F79 8706        PFRNT    LDW    CORESTART
3F7A 3762                 STB    /FRNT+4
3F7B 0A08                 SRL    8
3F7C 3761                 STB    /FRNT+3
3F7D 8707                 LDW    COREND
3F7E B706                 SUB    CORESTART
3F7F A7F1                 ADD    =2
3F80 0A11                 SLL    1
3F81 77B4                 STW    BYCOUNT
3F82 0040                 SLM
3F83 97F3                 LDX    =0
3F84 5F5E        STAT9    LDB  * FRNT
3F85 03C6                 DOT    X'C',6
3F86 02C0                 DIN    X'C',0
3F87 0A17                 SLL    7
3F88 0820                 SAM
3F89 1786                 JMP    $-3
3F8A 0401                 IXS    1
3F8B 0A10                 NOP
3F8C 0140                 CXA
3F8D F7F0                 CMW    =12
3F8E 0860                 SEQ
3F8F 1784                 JMP    STAT9
3F90 0050        PDMP     SGM
3F91 8706                 LDW    CORESTART
3F92 0A11                 SLL    1
3F93 0130                 CAX
3F94 5800        STAT10   LDB  * 0
3F95 03C6                 DOT    X'C',6
3F96 02C0                 DIN    X'C',0
3F97 0A17                 SLL    7
3F98 0820                 SAM
3F99 1796                 JMP    $-3
```

```
                ' LOAD AND DUMP PROGRAM
3F9A 0401                      IXS    1
3F9B 0A10                      NOP
3F9C 0140                      CXA
3F9D 0A01                      SRL    1
3F9E B7EC                      SUB    =1
3F9F F707                      CMW    COREND
3FA0 0860                      SEQ
3FA1 1794                      JMP    STAT10
3FA2 97F2       PTRAIL         LDX    =200
3FA3 0100       STAT11         CLR
3FA4 03C6                      DOT    X'C',6
3FA5 02C0                      DIN    X'C',0
3FA6 0A17                      SLL    7
3FA7 0820                      SAM
3FA8 17A5                      JMP    $-3
3FA9 0501                      DXS    1
3FAA 17A3                      JMP    STAT11
3FAB 03C0                      DOT    X'C',0
3FAC 0000                      HLT
3FAD 0080                      SMB    EXEC
3FAE 1020                      JMP    EXEC
3FAF 8A00       FRNT           D      X'8A00'
3FB0 0200                      D      X'0200'
3FB1 108D                      D      X'108D'
3FB2 0000                      D      0
3FB3 008A                      D      X'8A'
3FB4 0000       BYCOUNT        D      0
     3FB5       FRNTE          EQU    $
```

```
                207 * LOAD AND DUMP PROGRAM
       3FB5     208 MAGDMP   EQU   $
  3FB5 0050     209          SGM
  3FB6 0390     210          DOT   9,0
  3FB7 0290     211          DIN   9,0
  3FB8 0A43     212          SRC   3
  3FB9 0820     213          SAM
  3FBA 17D9     214          JMP   RING
  3FBB 976B     215          LDX   FORTY
  3FBC 0100     216          CLR
  3FBD 0393     217          DOT   9,3
       3FBE     218 MAGWWT   EQU   $
  3FBE 0290     219          DIN   9,0
  3FBF 0AC1     220          SRC L 1
  3FC0 0820     221          SAM
  3FC1 17BE     222          JMP   MAGWWT
  3FC2 5800     223          LDB * 0
  3FC3 039F     224          DOT   9,15
  3FC4 0401     225          IXS   1
  3FC5 17C7     226          JMP   MAGWXT
  3FC6 17BE     227          JMP   MAGWWT
       3FC7     228 MAGWXT   EQU   $
  3FC7 0290     229 MAGWEW   DIN   9,0
  3FC8 0AC1     230          SRC L 1
  3FC9 0820     231          SAM
  3FCA 17C7     232          JMP   MAGWEW
  3FCB 0100     233          CLR
  3FCC 0390     234          DOT   9,0
  3FCD 0004     235          HLT   4
  3FCE 0080     236          EXEC
S 3FCF 2020
                237
```

```
                 ' LOAD AND DUMP PROGRAM
3FD0 8DFF        MESS     D      X'8DFF'
3FD1 FF8A                 D      X'FF8A'
3FD2 D7D2                 TEXT   'WRITE RING'
3FD3 C9D4
3FD4 C5A0
3FD5 D2C9
3FD6 CEC7
3FD7 8DFF                 D      X'8DFF'
3FD8 FF8A        MESE     D      X'FF8A'
3FD9 0040        RING     SLM
3FDA 03E0                 DOT    X'E',0
3FDB 03EA                 DOT    X'E',X'A'
3FDC 97F3                 LDX    =0
3FDD 5FA0        RINGA    LDB  * MESS
3FDE 03EE                 DOT    X'E',X'E'
3FDF 02E0                 DIN    X'E',X'0'
3FE0 0A17                 SLL    7
3FE1 0820                 SAM
3FE2 17DF                 JMP    $-3
3FE3 0401                 IXS    1
3FE4 0000                 HLT
3FE5 0140                 CXA
3FE6 F7F4                 CMW    =18
3FE7 0860                 SEQ
3FE8 17DD                 JMP    RINGA
3FE9 03E0                 DOT    X'E',X'0'
3FEA 0000                 HLT
3FEB 1700                 JMP    BEGIN
                          END

3FEC 0001
3FED 0004
3FEE 0005
3FEF 000B
3FF0 000C
3FF1 0002
3FF2 00C8
3FF3 0000
3FF4 0012


NO ERRORS
```

```
BEGIN     3F00  BYCOUNT   3FB4  CHAR      3F53  COREND    3F07
CORESTAR  3F06  COUNT     3F51  DUMP      3F6C  ELEVEN    3F29
EXEC      0020  FIVE      3F23  FORTY     3F6B  FOUR      3F1B
FRNT      3FAF  FRNTE     3FB5  FRONT     3F14  INDEX     3F52
LAST      3F54  LEADER    3F0B  LOAD      3F03  MAGDMP    3FB5
MAGLOAD   3F55  MAGRWT    3F5A  MAGRXT    3F66  MAGWEW    3FC7
MAGWWT    3FBE  MAGWXT    3FC7  MESE      3FD8  MESS      3FD0
PDMP      3F90  PFRNT     3F79  PLEAD     3F6E  PLOAD     3F08
PTRAIL    3FA2  RING      3FD9  RINGA     3FDD  SETADDR   3F37
STAT1     3F0C  STAT10    3F94  STAT11    3FA3  STAT2     3F1D
STAT3     3F3E  STAT4     3F4A  STAT8     3F71  STAT9     3F84
TWELVE    3F2F
PAS?
```

I/O UNIT TEST PROGRAM

Unnecessary instructions were stripped from the program and it was relocated in the top end of memory (X' 3FF5') to allow use of a contiguous space it had divided.

The functions displayed via sense switch settings remain :

SENSE SWITCH	SENSOR WORD (with dial set on ACL)
Ø up	1
1 up	2
2 up	3
3 up	4

Because the program was short and obvious, the condensed version should now be even more obvious and therefore no flow chart is included.

```
       I/O UNIT TEST PROCEDURE                                          PAGE    1

                  'I/O UNIT TEST PROCEDURE
                  *
                  * VERSION OF 12/15/76    JGB
                  * MODIFIED TO FIT ABOVE THE LOAD DUMP ROUTINE
                  *
3FF5 008F                   SMB   $
     3FF6         AGAIN     EQU   $
3FF6 08C0                   SS0
3FF7 02F6                   DIN   15,6      SENSOR WORD 1
3FF8 08D0                   SS1
3FF9 02F5                   DIN   15,5      SENSOR WORD 2
3FFA 08E0                   SS2
3FFB 02F3                   DIN   15,3      SENSOR WORD 3
3FFC 08F0                   SS3
3FFD 02F7                   DIN   15,7      SENSOR WORD 4
3FFE 17F6                   JMP   AGAIN
                            ORIG  X'3FFF'
                            END


NO ERRORS

       I/O UNIT TEST PROCEDURE                                          PAGE    2

AGAIN     3FF6
PAS?
```


*U.S. GOVERNMENT PRINTING OFFICE : 1977 O-731-047/26

FHWA

R&D